



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV MIKROELEKTRONIKY

DEPARTMENT OF MICROELECTRONICS

## MONITOROVACÍ SYSTÉM TESTOVACÍHO ZAŘÍZENÍ ELEKTRICKÝCH MOTORŮ

MOTOR TEST PLATFORM MONITORING SYSTEM

### BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

### AUTOR PRÁCE

AUTHOR

Jan Klecker

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Michal Pavlík, Ph.D.

BRNO 2020

# Bakalářská práce

bakalářský studijní program **Mikroelektronika a technologie**

Ústav mikroelektroniky

**Student:** Jan Klecker

**ID:** 203411

**Ročník:** 3

**Akademický rok:** 2019/20

**NÁZEV TÉMATU:**

## Monitorovací systém testovacího zařízení elektrických motorů

### POKYNY PRO VYPRACOVÁNÍ:

Navrhnete systém měření teploty testovacího zařízení určeného pro zatěžování elektrických motorů u firmy NXP Semiconductor. Provedte analýzu systému z hlediska zdrojů a šíření tepla, navrhnete vhodné snímače teploty a jejich umístění na testovací stoličce. Zpracování informací o teplotě realizujete s použitím mikrokontroléru firmy NXP. Práce bude zaměřena na realizaci hardwarového a softwarového řešení.

### DOPORUČENÁ LITERATURA:

Podle pokynů vedoucího práce.

**Termín zadání:** 3.2.2020

**Termín odevzdání:** 8.6.2020

**Vedoucí práce:** Ing. Michal Pavlík, Ph.D.

**Konzultant:** Ing. Jaroslav Lepka, Ph.D., NXP Semiconductors

**doc. Ing. Jiří Háze, Ph.D.**  
předseda rady studijního programu

### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení částí druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **ABSTRAKT**

Obsahem této práce bude navrhnout systém pro měření teploty kritických částí testovací stolice pro měření vlastností pohonů s elektrickými motory firmy NXP. V rámci teoretické části bude analyzováno měření teploty pomocí teplotních senzorů a jejich připojení k mikrokontroléru. V praktické části pak bude realizováno samotné zařízení na měření a zpracování teploty. Jádrem celého zařízení bude mikrokontrolér z rodiny Kinetis MKVxx firmy NXP.

## **KLÍČOVÁ SLOVA**

Měření teploty, elektrický motor, 1-Wire sběrnice, I2C, NXP Kinetis, mikrokontrolér, jazyk C

## **ABSTRACT**

The content of this thesis is to design system for temperature measurement of critical parts of NXP electric engine testbench. In the theoretical part will be analysed temperature measurement by temperature sensors and its connecting to microcontroller. The practical part is focused on realization of device for temperature measurement. This device will be based on NXP microcontroller from Kinetis MKVxx family.

## **KEYWORDS**

Temperature measurement, electric engine, 1-Wire bus, I2C, NXP Kinetis, microcontroller, C language

KLECKER, J. Monitorovací systém testovacího zařízení elektrických motorů. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav mikroelektroniky, 2020. 42 s., 3 s. příloh. Bakalářská práce. Vedoucí práce: Ing. Michal Pavlík, Ph.D.

## Prohlášení autora o původnosti díla

**Jméno a příjmení studenta:** *Jan Klecker*

**VUT ID studenta:** *203411*

**Typ práce:** *Bakalářská práce*

**Akademický rok:** *2019/2020*

**Téma závěrečné práce:** *Monitorovací systém testovacího zařízení elektrických motorů*

*Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.*

*Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.*

V Brně dne: 6. června 2020

.....  
Podpis autor

# PODĚKOVÁNÍ

*Děkuji mému konzultantovi ve firmě NXP Ing. Jaroslavovi Lepkovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další velice cenné a praktické rady při zpracování mé bakalářské práce. V neposlední řadě děkuji mému vedoucímu práce Ing. Michalovi Pavlíkovi, Ph.D. za odborné konzultace a praktické rady.*

# OBSAH

<b>Úvod</b>	<b>1</b>
<b>1 TEORETICKÁ ČÁST</b>	<b>2</b>
1.1 TESTOVACÍ STANICE ELEKTRICKÝCH MOTORŮ .....	2
1.2 TEPLOTNÍ SENZORY .....	2
1.2.1 Analogové senzory .....	2
1.2.2 Digitální senzory .....	3
1.2.3 Porovnání analogových a digitálních senzorů .....	3
1.2.4 Typy teplotních senzorů.....	4
1.2.5 Výběr vhodného senzoru .....	7
1.3 TEPLOTNÍ SENZOR DS18B20 .....	8
1.3.1 Základní parametry .....	8
1.3.2 Vnitřní uspořádání .....	9
1.3.3 Ovládání.....	10
1.4 1-WIRE SBĚRNICE.....	11
1.4.1 Napájení slave zařízení .....	11
1.4.2 Princip ovládání .....	12
1.4.3 Zápis.....	12
1.4.4 Čtení.....	12
1.4.5 Identifikace zařízení.....	13
1.5 VÝBĚR MIKROKONTROLÉRU .....	13
1.5.1 Mikrokontrolér KV46 .....	14
1.5.2 Periferie eFlexPWM .....	16
1.5.3 Periferie PIT .....	17
1.5.4 Periferie I2C.....	17
1.6 VÝBĚR FUNKČNÍCH BLOKŮ .....	18
1.6.1 Ukládání hodnot.....	18
1.6.2 RTC.....	19
1.6.3 Displej.....	19
1.7 I2C SBĚRNICE .....	19
1.7.1 Komunikace .....	20

1.8	VÝVOJOVÉ NÁSTROJE .....	20
1.8.1	Vývojové prostředí .....	20
1.8.2	Program FreeMASTER .....	20
<b>2</b>	<b>PRAKTICKÁ ČÁST - HARDWARE</b>	<b>21</b>
2.1	NAPÁJECÍ ČÁST .....	21
2.2	OSTATNÍ ČÁSTI.....	22
<b>3</b>	<b>PRAKTICKÁ ČÁST – SOFTWARE</b>	<b>26</b>
3.1	BEZPEČNÉ MAZÁNÍ BITŮ .....	27
3.2	NÁVRH SOFTWARE PRO 1-WIRE SBĚRNICI.....	28
3.2.1	Nevhodnost oficiální knihovny .....	28
3.2.2	Řízení sběrnice periférií PWM .....	29
3.3	NÁVRH SOFTWARE PRO I2C SBĚRNICI .....	32
3.4	NÁVRH MĚŘICÍHO SOFTWARE .....	34
3.5	UŽIVATELSKÉ ROZHRAŇÍ .....	35
<b>4</b>	<b>ZÁVĚR</b>	<b>37</b>
	<b>Literatura</b>	<b>38</b>
	<b>Seznam symbolů, veličin a zkratk</b>	<b>40</b>
	<b>Seznam obrázků</b>	<b>40</b>
	<b>Seznam tabulek</b>	<b>42</b>
<b>A</b>	<b>návrh zařízení</b>	<b>43</b>
A.1	Obvodové zapojení měřicí desky.....	43
A.2	DPS měřicí desky – strana TOP .....	44
A.3	DPS měřicí desky – strana BOTTOM .....	44
A.4	Osazovací plán – strana TOP .....	45
A.5	Osazovací plán – strana BOTTOM .....	45



# ÚVOD

Testovací stolice elektrických motorů slouží k testování algoritmů a aplikací určených k řízení elektrických pohonů. Při testování se často testují mezní podmínky, v důsledku čehož může dojít k nenadálému nárůstu teploty nejen motoru, ale i kritických částí testovací stolice. Proto je důležité mít přehled o teplotě jednotlivých částí testovací stolice včetně testovaného motoru a na změny teploty reagovat odpovídajícím způsobem. Vzhledem k možnostem budoucího rozšíření teplotního monitorovacího systému o další funkce, jako je například řízení výkonu, je vhodné použít mikrokontrolér určený k řízení elektromotorů. Taktéž je potřeba zvolit vhodnou strukturu řídicího softwaru pro komunikaci se sběrnici a samotné měření.

V teoretické části bude rozebrána problematika měření teploty pomocí teplotních senzorů a vybrán vhodný senzor pro tuto aplikaci. Dále je v této části vybrán vhodný mikrokontrolér a rozepsány jeho parametry.

Praktická část je rozdělena do dvou kapitol – hardwarová a softwarová. V kapitole zabývající se hardwarem budou popsány jednotlivé části měřicí desky ilustrované jejich schémata. V kapitole zabývající se softwarem bude ukázán systematický přístup k návrhu softwaru a dále pak členění projektu. Následně bude podrobně rozebrán návrh a implementace ovládacího softwaru pro sběrnici. Poté bude navržena konstrukce pro samotné měření a zpracování teploty s uživatelským rozhraním.

# 1 TEORETICKÁ ČÁST

## 1.1 TESTOVACÍ STANICE ELEKTRICKÝCH MOTORŮ

Testovací stolice pro měření a analýzu elektrických motorů se ve firmě NXP používají k testování metod řízení elektrických pohonů. Motory jsou umístěny na kovové konstrukci, umožňující seřizování polohy motoru nezávisle ve třech osách x, y, z.



Obr. 1: Testovací stolice s elektromotorem

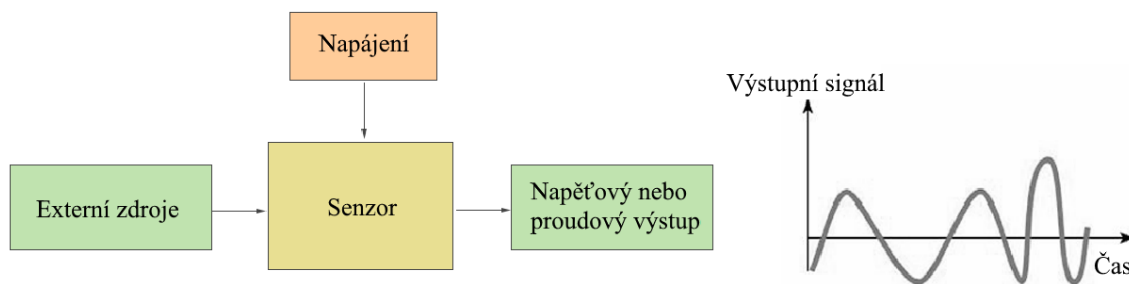
## 1.2 TEPLOTNÍ SENZORY

Teplota patří k nejčastěji měřeným veličinám v prostředí. V dnešní době je přesné měření teploty velice důležité ve všech odvětvích průmyslu – medicínské aplikace, biologický výzkum, letecký průmysl. S rozvojem mikrokontrolérů a embedded systémů se rozvinuly také teplotní senzory. Měření teploty může být realizováno přímým kontaktem senzoru a zdroje tepla, nebo vzdáleně, kdy senzor měří vyzařovanou energii ze zdroje tepla.

Teplotní senzory jsou obecně zařízení, která sbírají informaci o teplotě určitého zdroje a převádí ji buď na analogovou, nebo na digitální hodnotu.

### 1.2.1 Analogové senzory

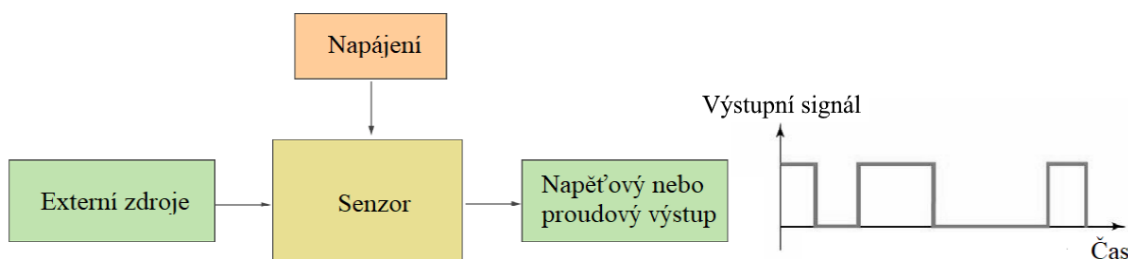
Analogové teplotní senzory snímají teplotu ze zdroje tepla, kterou na svém výstupu reprezentují analogovým signálem. Výstupy těchto senzorů jsou připojovány do AD převodníků, popřípadě do měřicích můstků.



Obr. 2: Blokové schéma analogového senzoru [1]

### 1.2.2 Digitální senzory

Digitální teplotní senzory snímají teplotu ze zdroje tepla, kterou na svém výstupu reprezentují diskretními hodnotami signálu – log. 0 nebo log. 1. Tyto senzory jsou k mikrokontroléru nejčastěji připojeny pomocí sběrnic – I2C, 1-Wire a další.



Obr. 3: Blokové schéma digitálního senzoru [1]

### 1.2.3 Porovnání analogových a digitálních senzorů

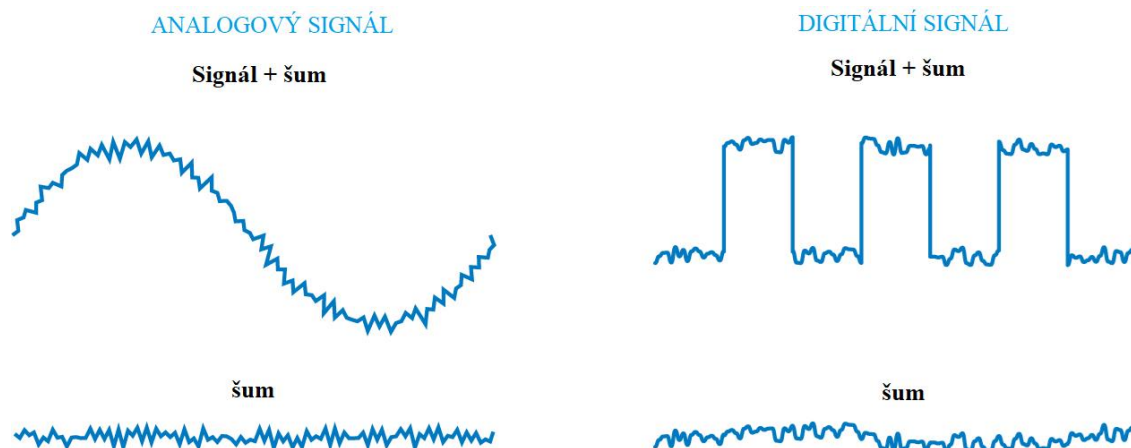
Nejprve je potřeba rozlišit dva pojmy popisující vlastnosti teplotních senzorů, a to rozlišení a přesnost.

Přesnost, v anglické literatuře „accuracy“, udává, jak blízko se měřená hodnota skutečně nachází od té reálné. Udává se v jednotkách  $\pm^\circ\text{C}$ . Údaj  $\pm 1^\circ\text{C}$  tedy znamená, že se měřená teplota ve skutečnosti může nacházet kdekoli v daném intervalu.

Rozlišení, v anglické literatuře „resolution“, udává, jakou nejmenší změnu teploty je senzor schopen zaznamenat. Údaj rozlišení  $0,1^\circ\text{C}$  tedy znamená, že jestliže je změna teploty menší než tato hodnota, nebude senzorem zaznamenána.

Analogové senzory jsou poměrně snadno ovlivnitelné šumem, což způsobuje chyby ve výstupním signálu. Mají větší rozlišení a mohou tak být použity pro větší teplotní rozsah. Problém může nastat u delšího vedení, kdy se projevuje jeho odpor a průmyslové rušení. [2]

Oproti tomu digitální senzory jsou vůči šumu odolnější, ale mají menší teplotní rozsah. Nevýhodou je, že přenosová rychlost je ovlivňována samotnou sběrnicí, což při připojení více senzorů může být limitující faktor. Obvykle mají i větší přesnost než analogové senzory.



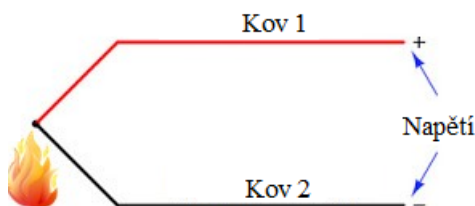
**Obr. 4: Šum v analogovém a digitálním signálu [2]**

Na Obr. 4 je ukázáno ovlivnění analogového a digitálního signálu šumem. Z obrázku je patrné, proč u analogového signálu způsobuje šum větší nepřesnosti. Jelikož analogový signál je zpracováván analogově, tak šum ovlivňuje převáděnou hodnotu signálu, čímž je způsobena nepřesnost měření. Oproti tomu digitální signál má poměrně velký odstup log. 1 od log. 0, tudíž je proti rušení odolnější.

## 1.2.4 Typy teplotních senzorů

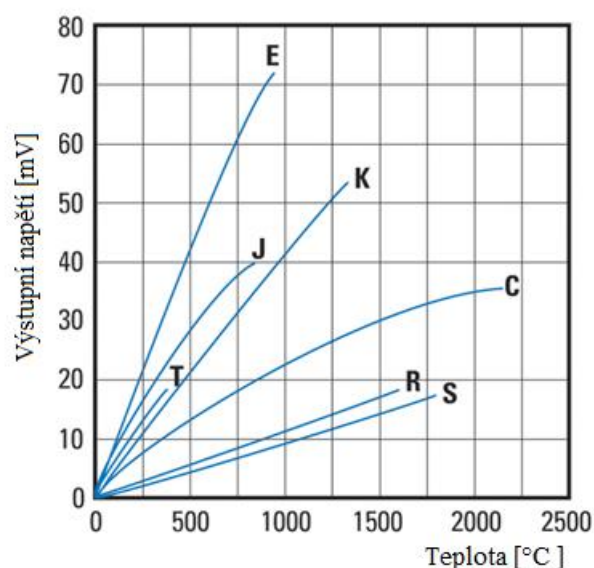
### Termočlánek

Termočlánky patří k nejpoužívanějším teplotním senzorům. Jsou založeny na termoelektrickém jevu (Seebeckův jev). Ten vzniká tak, že elektrony v teplejší části vodiče mají větší energii, tudíž difundují do chladnějších míst více než elektrony z chladnějších míst do teplejších. Termočlánek se skládá ze dvou různých vodičů či polovodičů. Jestliže mají tyto spoje rozdílné teploty, protéká tímto obvodem elektrický proud. Jestliže obvod v libovolném místě rozpojíme, na nově vzniklých svorkách vznikne napětí [3].



**Obr. 5: Princip termočlánku [3]**

Podle použitých materiálů se termočlánky dělí na více typů – například typ J, který se skládá ze železa a konstantanu. Hlavní výhodou termočlánků je velký rozsah teplot, konkrétně typ B má teplotní rozsah 0 až 1600 °C.



Obr. 6: Závislost generovaného napětí na teplotě pro různé druhy termočlánků [4]

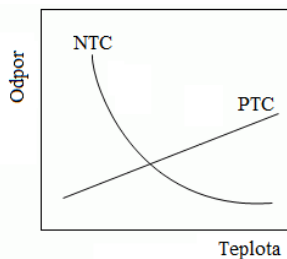
## RTD

RTD, neboli „Resistance Temperature Detector“, je teplotní senzor, jehož odpor se mění v závislosti na teplotě. Nejčastěji jsou vyrobeny z platiny, vyskytují se také varianty s niklem či mědí. Mají kladný teplotní koeficient (PTC) a jejich výstup má velice lineární charakter, tudíž umožňují přesné měření teploty. Jak již bylo zmíněno, nejčastěji jsou vyrobeny z platiny, stejně jako nejrozšířenější typ, PT100. Tento senzor má hodnotu odporu 100  $\Omega$  při teplotě 0 °C. [5]

Aby bylo možné měřit teplotu, musí přes senzor protékat proud, čímž vzniká úbytek napětí, který lze vyhodnotit.

## Termistor

Stejně jako u RTD, tak i u termistory pracují na principu měnící se hodnoty odporu v závislosti na teplotě. Rozlišují se dva typy termistorů, PTC (Positive Temperature coefficient) a NTC (Negative Temperature Coefficient). U PTC termistoru s rostoucí teplotou roste i jeho hodnota odporu, u NTC naopak s rostoucí teplotou hodnota odporu klesá.



Obr. 7: Závislost odporu na teplotě pro PTC a NTC [6]

Termistory jsou cenově dostupné a snadno se implementují do obvodů. Z Obr.7 je patrné, že malá změna teploty vyvolá poměrně značnou změnu odporu, což dělá z termistorů velice citlivá zařízení. Protože moc dobře nepracují v příliš vysokých

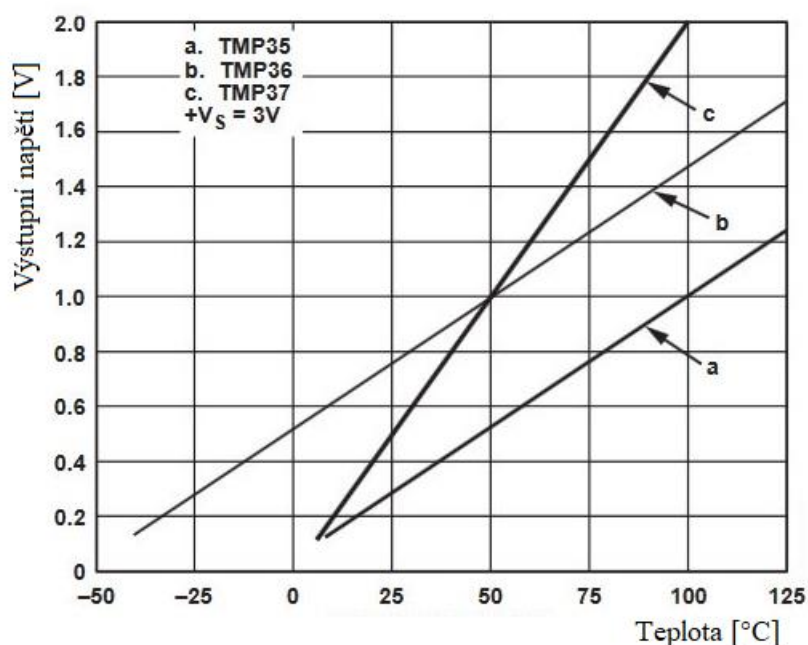
a nízkých teplotách, využívají se k měření teploty kolem stanoveného bodu. Používají se například v automobilovém průmyslu k měření teploty oleje, nebo v teplotně ochranných obvodech, kde při nárůstu teploty dojde ke změně jejich odporu – ten je pak snímán a indikuje nárůst teploty. [6]

### IC teplotní senzory [7]

Jedná se o polovodičové integrované obvody, které obsahují senzor teploty a kompenzační obvody v jednom pouzdře. Jsou vyráběny ve více variantách, mohou být s analogovými snímacími obvody a s proudovým/napěťovým výstupem nebo s analogovými snímacími obvody a s digitálními výstupy a s kontrolními registry, což z nich dělá ideální řešení pro obvody s mikrokontroléry.

#### Analogové

Namísto použití termistoru s dalšími externími součástkami nabízí alternativu analogové IO teplotní senzory. Například TMP36 od firmy Analog Devices zaručuje téměř lineární výstupní napětí, s teplotním rozsahem -40 až 125 °C.



Obr. 8: Závislost výstupního napětí na teplotě senzorů firmy Analog Devices [7]

#### Digitální

Digitální senzory zjednodušují výsledné zapojení, jelikož odpadá nutnost dalších externích komponent – A/D převodník je zakomponovaný už v samotném senzoru. Komunikace probíhá nejčastěji přes sběrnice – I2C, 1-Wire atd. Velice rozšířeným digitálním teplotním senzorem je DS18B20 od firmy Maxim Integrated, které má přesnost  $\pm 0.5$  °C a teplotní rozsah od -55 do +125 °C.

Tabulka 2: Srovnání teplotních senzorů

Typ senzoru	Typický teplotní rozsah [°C]	Přesnost [± °C]	Výhody	Nevýhody
<b>Termistor</b>	-40 až 125	1	Levný. Malé rozměry odolný.	Nelineární výstup. Pomalá odezva.
<b>Termočlánek</b>	-200 až 1450	2	Vysoké rozlišení. Malé rozměry. Velký rozsah teplot.	Nutná kalibrace. Potřeba měřit dva body.
<b>RTD</b>	-260 až 850	1	Lineární výstup Přesnost	Dražší než termistor. Křehký.
<b>Analogový IO (TMP36)</b>	-40 až 125	2	Lineární výstup. Jednoduchý na použití.	Dražší než termistor. Limitovaný teplotní rozsah.
<b>Digitální IO (DS18B20)</b>	-55 až 125	0,5	Ideální pro mikrokontroléry Lineární výstup.	K činnosti potřebuje mikrokontrolér. Dražší než termistor. Limitovaný teplotní rozsah.

### 1.2.5 Výběr vhodného senzoru

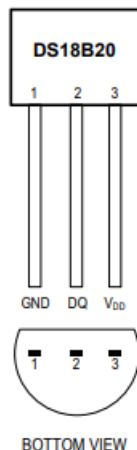
Přehled teplotních senzorů z podkapitol výše tvoří teoretický základ pro výběr vhodného senzoru teploty pro systém měřící teplotu.

Nejprve je nutné se rozhodnout, zda-li použít analogový či digitální senzor. Vzhledem k tomu, že testovací stolice elektrického motoru se nachází v laboratoři, kde se testují další zařízení, je tento prostor zatížen šumem. Navíc samotné senzory jsou umístěny na elektrickém motoru a v jeho blízkosti a také v blízkosti měniče pro řízení motoru, který je sám o sobě zdrojem rušení. Z těchto důvodů byl upřednostněn digitální senzor před analogovým.

Nejčastějšími sběrnici pro teplotní senzory jsou I2C, SPI a 1-Wire. Při měření teploty na elektrickém motoru by samotné rozmístění senzorů nemělo být limitováno délkou přívodního kabelu senzoru. Sběrnice I2C a SPI nejsou příliš vhodné pro komunikaci na delší vzdálenosti (jednotky metrů). Z těchto důvodů byla vybrána sběrnice 1-Wire a teplotní senzor DS18B20 od firmy Maxim Integrated.

## 1.3 TEPLITNÍ SENZOR DS18B20

Senzor DS18B20 vyráběný firmou Dallas Semiconductor je digitální IC teplotní senzor s rozlišením od 9 do 12 bitů. Tyto senzory komunikují pomocí 1-Wire sběrnice, která vyžaduje pouze jeden datový vodič a zem GND. Mají možnost parazitního napájení, kdy je senzor napájen přímo z datového vodiče. Senzor se vyrábí ve třech pouzdrech – SO,  $\mu$ SOP a TO-92.



Obr. 9: Rozložení pinů [8]

### 1.3.1 Základní parametry

Teplotní senzor DS18B20 měří v teplotním rozsahu od  $-55$  do  $+125$  °C. Běžně udávaná přesnost  $\pm 0,5$  °C však platí pouze v rozsahu teplot od  $-10$  do  $+85$  °C. Senzor umožňuje nastavit rozlišení 9, 10, 11 a 12 bitů, což koresponduje s rozlišením teploty 0,5 °C, 0,25 °C, 0,125 °C a 0,0625 °C. Umožňuje také takzvaný Parasitic power mode, při kterém jsou potřeba k činnosti pouze 2 vodiče – datový DQ a zem GND. Důležité parametry senzoru jsou popsány v Tabulce 3.

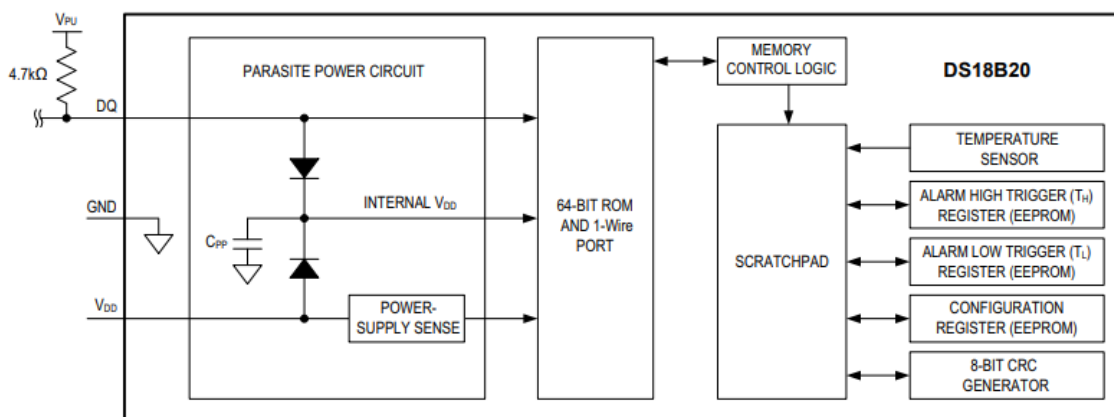
Tabulka 3: Výběr základních parametrů senzoru DS18B20 [8]

Parametr	Symbol	Podmínky	min	max	Jednotka
Napájecí napětí	$V_{DD}$		3	5,5	V
Napájecí napětí pullup rezistoru	$V_{PU}$	Parazitní napájení	3	5,5	V
		Normální napájení	3	VDD	
Chyba měření teploty	$t_{ERR}$	10 až +85 °C		$\pm 0,5$	°C
		30 až 100 °C		$\pm 1$	
		55 až 125 °C		$\pm 2$	
Sink current	$I_L$	$V_{IO} = 0,4$ V	4		mA
Active current	$I_{DD}$	$V_{DD} = 5$ V	1	1,5	mA
Čas konverze teploty	$t_{conv}$	Rozlišení 9 bitů		93,75	ms
		Rozlišení 10 bitů		187,5	
		Rozlišení 11 bitů		375	
		Rozlišení 12 bitů		750	
Časový slot	$t_{slot}$		60	120	$\mu$ s
Čas zotavení	$t_{REC}$		1		$\mu$ s



### 1.3.2 Vnitřní uspořádání

Na Obr. 10 lze vidět zjednodušené uspořádání senzoru DS18B20. V levé části diagramu je vstupní část s piny DQ, GND a  $V_{DD}$ . Blok PARASITE POWER CIRCUIT zajišťuje napájení samotného senzoru. Lze si vybrat mezi napájením z vodiče  $V_{DD}$  nebo takzvaným parazitním napájením z kondenzátoru  $C_{PP}$ . Parazitní napájení je popsáno v kapitole zaměřené na 1-Wire sběrnici.



Obr. 10: Vnitřní uspořádání senzoru DS18B20 [8]

Na výstupu senzoru se nachází MOSFET tranzistor. Při vyslání log. 0 je tranzistor sepnut a spojí tak sběrnici se zemí GND. Při vyslání log. 1 není tranzistor sepnutý a logickou úroveň na sběrnici udržuje pull-up rezistor. V dalším bloku je obsažená paměť 64bitová paměť ROM a vstupně-výstupní PORT. PORT je propojený s pamětí SCRATCHPAD.

Paměť SCRATCHPAD je typu SRAM. Ta je dále propojena s pamětí EEPROM, kde jsou uloženy registry alarmu (TH a TL) a konfigurační registr. Dále je propojena s 8bitovým CRC generátorem a se samotným obvodem měřící teplotu.

BYTE 0 a BYTE 1 tvoří registr ukládající hodnotu teploty. Teplota je ukládána v jednotkách  $^{\circ}\text{C}$  ve formátu dvojkového doplňku. Po power-on resetu je v tomto registru uložena výchozí hodnota  $+85^{\circ}\text{C}$ .

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
LS BYTE	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>	2 <sup>-1</sup>	2 <sup>-2</sup>	2 <sup>-3</sup>	2 <sup>-4</sup>
	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
MS BYTE	S	S	S	S	S	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>

Obr. 11: Registr pro ukládání teploty [8]

BYTE 2 ( $T_H$ ) a BYTE 3 ( $T_L$ ) registry slouží k nastavení teplotních limitů. Jestliže se měřená teplota nachází mimo rozsah těchto limitů, v senzoru se nastaví Alarm flag. Podmínka se testuje po každém převodu teploty. Jestliže se teplota vrátí do nastaveného rozsahu, Alarm flag se vymaže. Master zařízení tento flag testuje pomocí příkazu Alarm Search. Tyto registry mohou být zkopírovány do paměti EEPROM.

BYTE 4 registr je konfigurační registr, který slouží k nastavení rozlišení měřené teploty. Pomocí bitů R0 a R1 můžeme nastavit čtyři rozlišení dle potřeby – 9, 10, 11 nebo 12 bitů. Tento registr může být taktéž zkopírován do paměti EEPROM.

BYTE 8 registr je pouze ke čtení a slouží k uložení CRC kód. CRC kód je počítán z dat uložených v paměti scratchpad.

### 1.3.3 Ovládání

Jak již bylo zmíněno, senzory DS18B20 se připojují k mikrokontroléru pomocí 1-Wire sběrnice. Samotnému popisu fyzického ovládání této sběrnice bude věnována podkapitola 1.4.2. Jestliže zařízení není ovládáno pomocí příkazů, nachází se v Idle stavu. V případě vyslání příkazu (převod teploty, čtení z paměti) se zařízení „vzbudí“, vykoná danou operaci a vrátí se zpátky do Idle stavu.

Senzor DS18B20 se ovládá pomocí 8bitových příkazů. Ty lze rozdělit do dvou skupin – ROM příkazy a funkční příkazy.

Tabulka 4: ROM příkazy [8]

Příkaz	Popis	Kód
Search Rom	Najde ROM kód všech zařízení připojených ke sběrnici	0xF0
Read Rom	Najde ROM kód zařízení za předpokladu, že jako jediné připojené ke sběrnici	0x33
Match Rom	Umožňuje adresovat dané zařízení	0x55
Skip Rom	Master zařízení může adresovat všechna zařízení na sběrnici	0xCC
Alarm Search	Umožňuje zjistit, zda-li byla překročena nastavená teplota	0xEC

Nejvíce využívané ROM příkazy jsou Match Rom a Skip Rom. Match Rom se využívá k adresaci jednoho daného zařízení. Následně za ním je vyslán funkční příkaz, například pro čtení paměti. Příkaz Skip Rom se využívá ve spojení s příkazem na převod teploty, kdy chceme převést teplotu u všech zařízení na připojených na sběrnici.

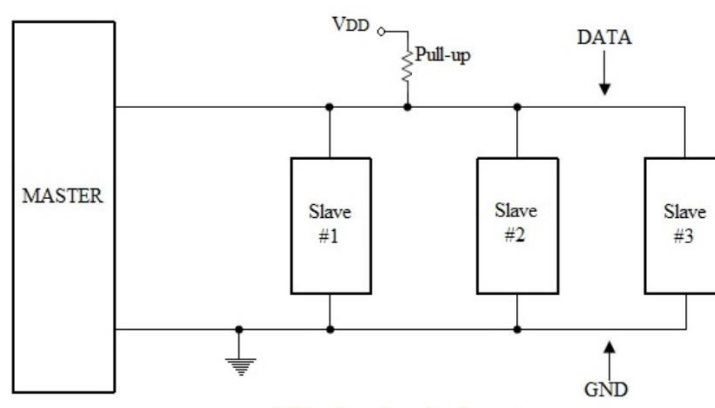
Tabulka 5: Funkční příkazy [8]

Příkaz	Popis	Kód
Convert T	Inicializuje teplotní převod senzoru	0x44
Write Scratchpad	Inicializuje zápis do paměti scratchpad	0x33
Read Scratchpad	Inicializuje čtení z paměti scratchpad	0xBE
Copy Scratchpad	Inicializuje kopírování TH, TL a konfiguračního registru z paměti scratchpad do paměti EEPROM	0x48
Recall E2	Inicializuje kopírování TH, TL a konfiguračního registru z paměti EEPROM do paměti scratchpad	0xB8
Read Power Supply	Inicializuje zjištění zdroje napájení	0xB4

Nejpoužívanějšími příkazy z této skupiny jsou Convert T a Read Scratchpad. Příkaz Convert T spustí převod teploty a následně uloží změřenou hodnotu do paměti scratchpad. Příkaz Read Scratchpad umožňuje čtení z paměti scratchpad od prvního do devátého bytu. V případě potřeby načtení jen určitých bytů lze čtení přerušit vyvoláním reset pulzu.

## 1.4 1-WIRE SBĚRNICE

1-Wire je asynchronní sběrnice navržena firmou Dallas Semiconductor. Sběrnice má jeden řídicí obvod (master), který řídí jedno či více ovládaných zařízení (slave). Obvykle je používána pro komunikaci s malými levnými zařízeními, jako jsou teplotní senzory či paměti EEPROM. Oproti konkurenční I<sup>2</sup>C sběrnici má pomalejší rychlost přenosu, ale lze ji použít na delší vzdálenosti. Všechny obvody jsou zapojeny jednak na společnou zem (GND), jednak paralelně na společný datový vodič (DQ). Tento datový vodič je připojen přes pull-up rezistor o hodnotě přibližně 5 k $\Omega$  na napájecí napětí a udržuje tak sběrnici ve stavu log. 1.



Obr. 12: Uspořádání 1-Wire sběrnice

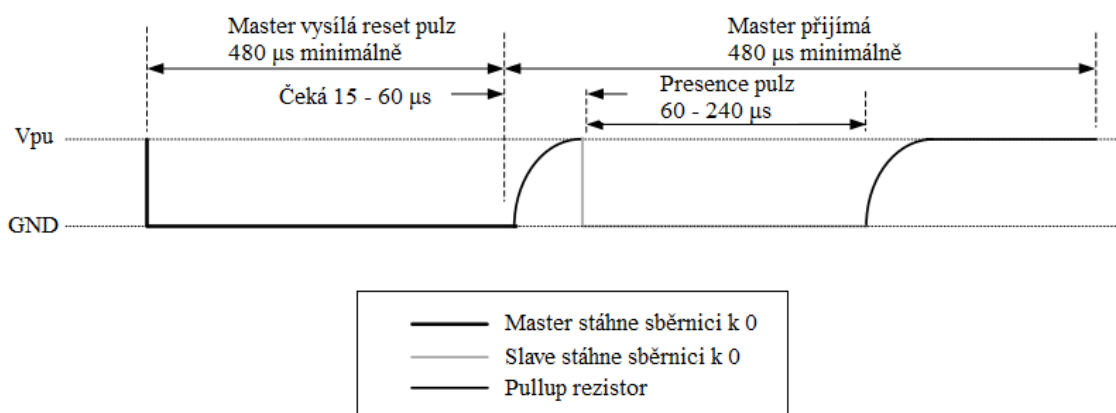
### 1.4.1 Napájení slave zařízení

1-Wire sběrnice dokáže fungovat ve dvou režimech. První a obvyklý režim fungování je pomocí tří vodičů – první pro napájení slave zařízení ( $V_{DD}$ ), druhý pro datovou komunikaci (DQ) a třetí pro společnou napájecí i datovou zem (GND). Ve druhém režimu je zařízení napájeno v takzvaném parazitním režimu. V tomto režimu jsou využívány pouze dva vodiče – datový (DQ) a zem (GND). Zařízení jsou napájena datovým vodičem, když je sběrnice v úrovni log 1. Toto napětí nabíjí interní kondenzátor (C<sub>pp</sub>), který napájí zařízení, když je sběrnice v úrovni log. 0. Tato metoda napájení z 1-Wire sběrnice je nazývána jako „parasit power.“

## 1.4.2 Princip ovládání

Jak již bylo zmíněno výše, jedná se o asynchronní sběrnici, tudíž není synchronizována pomocí hodinového signálu. Data jsou vysílána a přijímána v takzvaných časových slotech (time slot). Časový slot je dlouhý 60 až 120  $\mu\text{s}$  a v jeho průběhu je vyslán či přijat jeden bit. Komunikaci po sběrnici zahajuje vždy master vysláním reset pulzu – nejprve datový vodič stáhne do log. 0 a minimálně 480  $\mu\text{s}$  jej drží v této úrovni. Poté sběrnici uvolní a naslouchá (sběrnice je držena pull-up rezistorem v log. 1). Jestliže je na sběrnici připojené zařízení, tak detekuje vzestupnou hranu vyvolanou zařízením master a po prodlevě 15 – 60  $\mu\text{s}$  stáhne sběrnici na 60 – 240  $\mu\text{s}$  k log. 0.

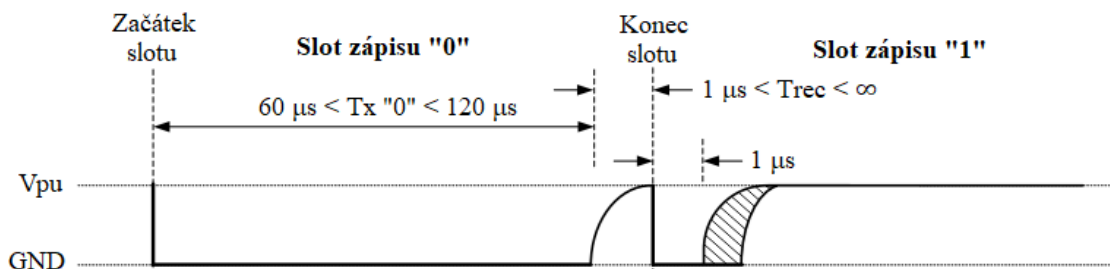
Jestliže se slave zařízení správně ohlásí, může master začít vysílat data. Jak již bylo uvedeno výše, data jsou vysílána v časových slotech. Mezi jednotlivými sloty by měla být minimálně 1  $\mu\text{s}$  mezera, kdy je sběrnice v klidu.



Obr. 13: Časový průběh reset pulzu [8]

## 1.4.3 Zápis

Rozlišujeme dva druhy zápisových slotů – zápis 1 a zápis 0. Tyto sloty slouží k tomu, aby master zapsal data do zařízení. Zápis log. 1 probíhá tak, že master stáhne sběrnici k log. 0 na 1 – 15  $\mu\text{s}$  a poté ji uvolní. K zápisu log. 0 master stáhne sběrnici k log. 0 a ponechá ji v tomto stavu po trvání celého časového slotu, tedy minimálně 60  $\mu\text{s}$ .

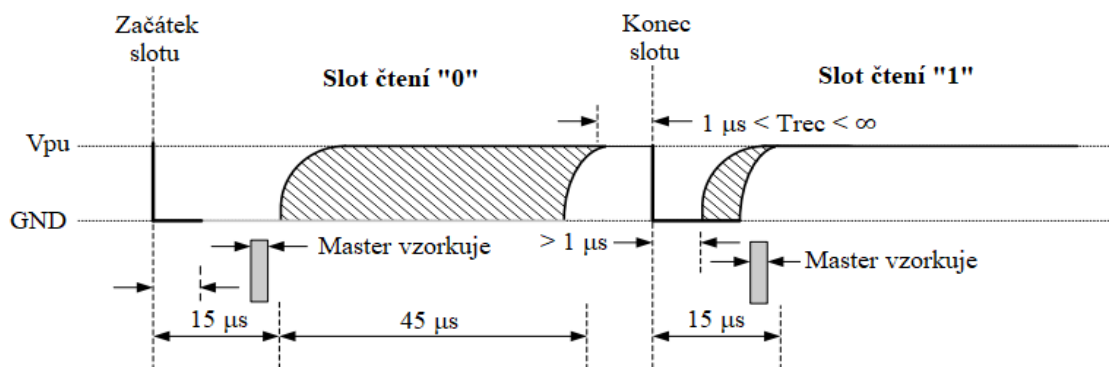


Obr. 14: Time sloty pro zápis log 0. a log. 1 [8]

## 1.4.4 Čtení

Čtecí sloty master inicializuje tak, že stáhne sběrnici k log 0. minimálně na 1  $\mu\text{s}$  a opět ji uvolní. Po tomto procesu může slave vyslat 1 bit. Pro zápis log. 1 na sběrnici

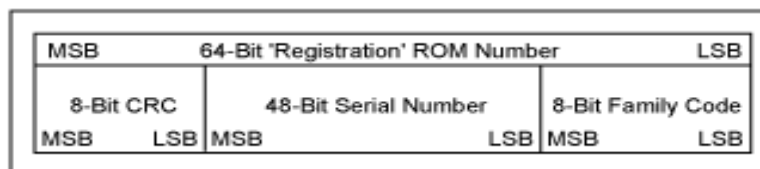
slave zařízení ponechá sběrnici v klidu, pro zápis log. 0 ji stáhne.



Obr. 15: Time slots pro čtení log. 0 a log. 1 [8]

### 1.4.5 Identifikace zařízení

V případě, že se na sběrnici vyskytuje pouze jediné zařízení, není potřebná žádná adresace. Jestliže je však na sběrnici připojeno více zařízení, je nutná adresace zařízení, se kterým se má navázat komunikace. Z tohoto důvodu obsahují 1-Wire zařízení ROM paměť, ve které je uloženo 64bitové unikátní číslo, pomocí kterého je možno jednotlivá zařízení na sběrnici od sebe odlišit. Tento 64bitový kód se skládá z typu zařízení (spodních 8 bitů), sériového čísla (48 bitů) a z CRC kódu (nejvyšších 8 bitů). Princip ovládání je pak následující: Po vyslání RESET pulzu je nutné vyslat příkaz Match ROM následovaný 64bitovým kódem zařízení. Tím je specifikováno, se kterým zařízením bude komunikováno – ostatní zařízení se uvedou do stavu vysoké impedance. Po příkazu Match ROM už mohou následovat jednotlivé funkční příkazy pro dané zařízení.

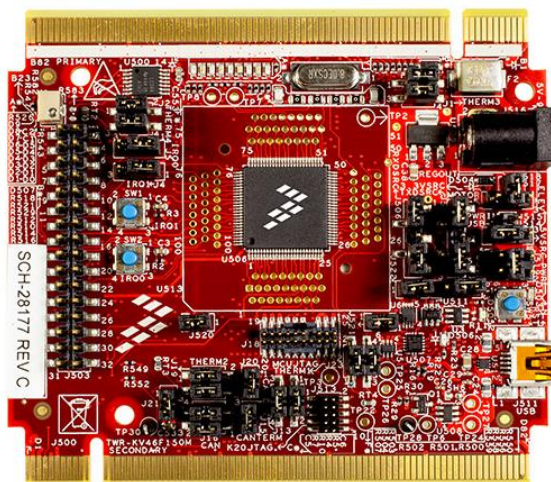


Obr. 16: 64bitové číslo uložené v ROM [8]

## 1.5 VÝBĚR MIKROKONTROLÉRU

V požadavku na zpracování tohoto projektu bylo použití vhodného mikrokontroléru z rodiny Kinetis KV4x od firmy NXP. Za předpokladu, že by šlo pouze o stanici měřící a vyhodnocující teplotu, bylo by použití tohoto mikrokontroléru zbytečné – vysoký výkon a velké množství periférií. Avšak blok měřící teplotu je zamýšlen jako část celku, ke kterému bude moci být připojeno například chlazení motoru, zatěžovacího zařízení či řízení jeho výkonu. S ohledem na tyto požadavky byl vybrán mikrokontrolér MKV46F256VLL16 na vývojové desce TWR-KV46150M, který patří do rodiny mikrokontrolérů Kinetis V. Platforma Tower umožňuje snadné propojení více vývojových desek pomocí konektorů po bocích desky (například vývojová deska pro řízení třífázových BLDC motorů). Výhodou této desky je rozhraní OpenSDA, což je

integrované debugovací rozhraní zajišťující komunikaci mezi počítačem (nebo jiné USB zařízení) a cílovým mikrokontrolérem pouze pomocí USB kabelu.

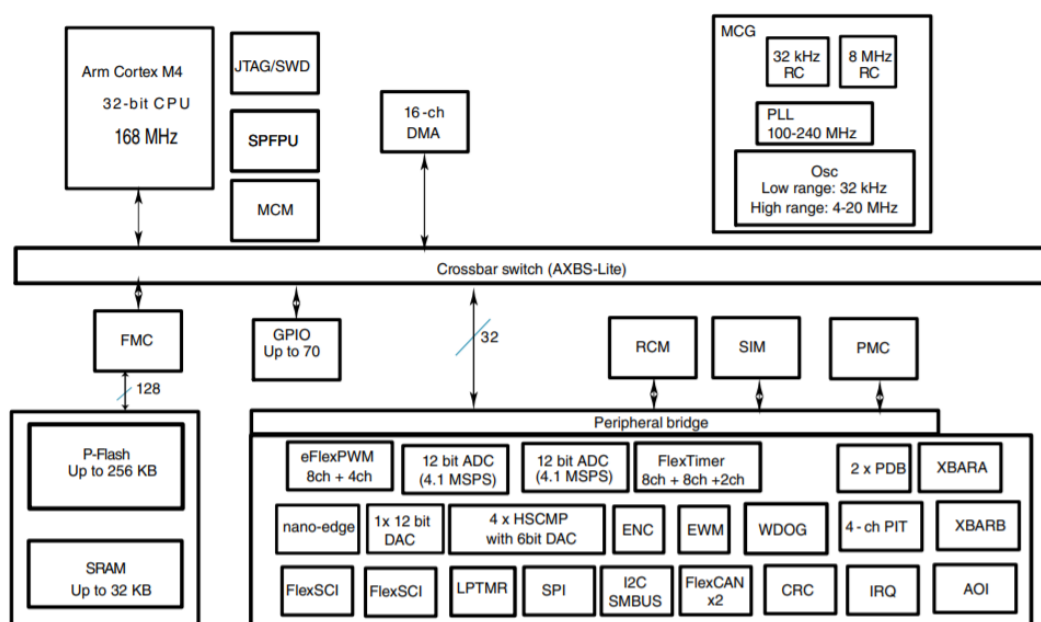


Obr. 17: Vývojová deska TWR-KV46150M [9]

### 1.5.1 Mikrokontrolér KV46

Rodina mikrokontrolérů Kinetis KV4x je součástí série mikrokontrolérů Kinetis V a je řešením pro výkonné a efektivní řízení elektrických motorů. Mikrokontrolér je postaven na jádře ARM®Cortex®-M4 s frekvencí do 168 MHz. Mezi hlavní funkce patří: duální 12bitový ADC převodník s 240 ns časem převodu, eFlexPWM modul s 312 ps rozlišením, delay block, dual FlexCan modul a 64 až 256 KB flash paměť.

Mikrokontroléry jsou dostupné v těchto provedeních pouzder: 48, 64 a 100 LQFP.



Obr. 18: Blokový diagram mikrokontroléru rodiny Kinetis V [10]

## **Výběr hlavních parametrů [10]:**

### **Paměti:**

- Do 256 KB programové paměti flash
- Do 32 KB paměti RAM

### **Systémové periferie:**

- 16kanálový DMA kontrolér
- Nezávisle časovaný watchdog

### **Clock:**

- 32 až 40 kHz nebo 3 až 32 MHz krystalový oscilátor
- Multipurpose Clock Generator (MCG) s fázovým a frekvenčním závěsem vztahujícím se buď k internímu nebo externímu zdroji hodinového signálu

### **Provozní charakteristika:**

- Napěťový rozsah: 1,71 až 3,6 V
- Teplotní rozsah: -40 až 105 °C

### **Vstupně-výstupní rozhraní:**

- General-purpose input/output

### **Komunikační rozhraní:**

- Dva Universal Asynchronous Receiver/Transmitter (UART) / FlexSCI moduly s programovatelným 8 nebo 9bitovým datovým formátem
- Jeden 16bitový SPI modul
- Jeden I2C modul
- Dva FlexCAN moduly

### **Analogové moduly:**

- Dva 12bitové ADC moduly
- Čtyři analogové komparátory (CMP) obsahující a 6bitový DAC převodník a programovatelný referenční vstup
- Jeden 12bitový DAC převodník

### **Timery:**

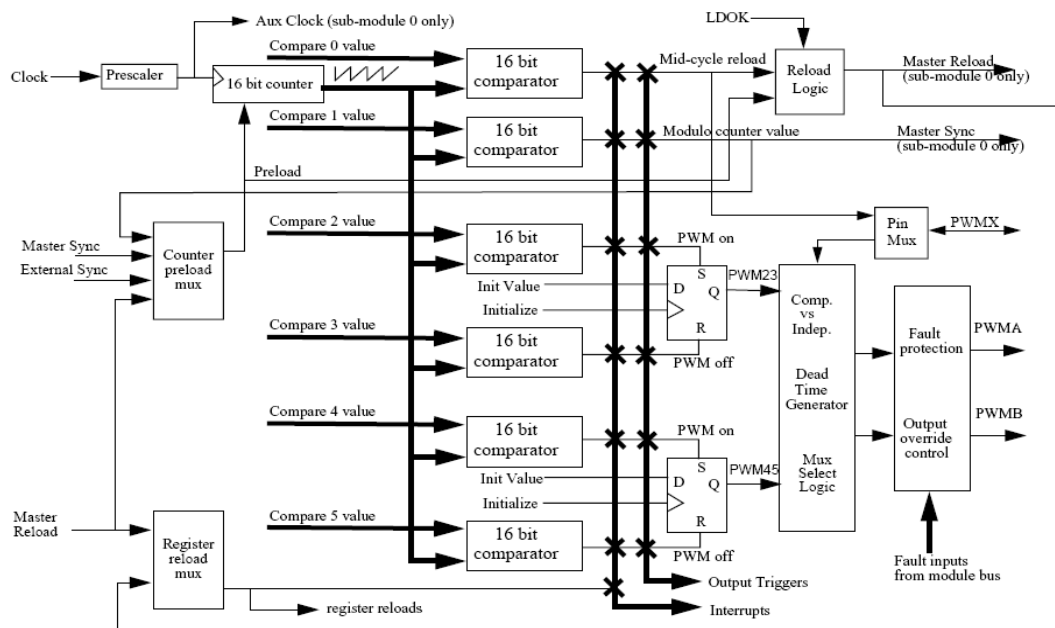
- Jeden eFlexPWM s 4 submoduly, s možností 12 PWM výstupů
- Dva 8kanálové FlexTimers (FTM0 a FTM3)
- Jeden 2kanálový FlexTimer (FTM1)
- Čtyři Periodic interrupt timers (PIT)
- Dva Programmable Delay Blocks (PDB)
- Quadrature Encoder/Decoder (ENC)
- Poměr mezi vstupní frekvencí časovače a frekvencí jádra je 1:2 jestliže

frekvence jádra je 168 MHz

- Poměr mezi vstupní frekvencí časovače a frekvencí jádra je 1:1 jestliže frekvence jádra je menší rovna 100 MHz

### 1.5.2 Periferie eFlexPWM

Jedná se o PWM periferii, která slouží ke komplexnímu a flexibilnímu řízení měničů pro řízení elektrických motorů. Skládá se ze čtyř nezávislých submodulů s nezávislými časovými základnami. Každý submodul disponuje dvěma nezávislými výstupy a jedním pomocným vstupem/výstupem.



Obr. 19: Vnitřní schéma periferie eFlexPWM [10]

Perioda a střída PWM je nastavována pomocí *Init* a *Value* registrů. Všechny tyto registry jsou bufferované, což zaručuje synchronizaci. Tato hodnota je zpropagována do stínového registru, když je bit *MCTRL[LDOK]* nastaven. Hodnoty těchto registrů jsou májí efekt až při dalším reloadu PWM.

**Initial Count Register:** 16bitová znaménková hodnota udává, od jakého čísla začíná čítač PWM čítat

**Value Registr 1:** 16bitová znaménková hodnota udává maximální hodnotu čítače

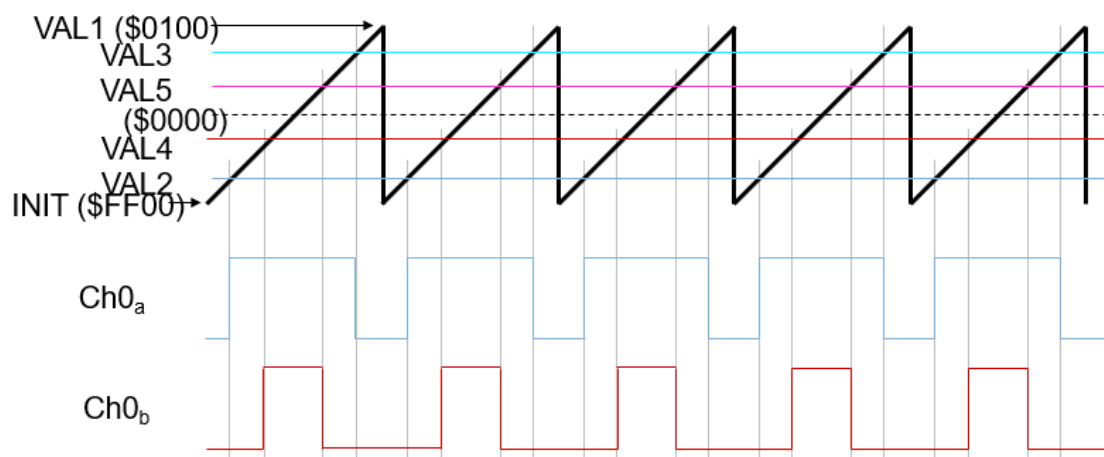
**Value Registr 2:** 16bitová znaménková hodnota udává hodnotu nastavující PWM23 do logické 1

**Value Registr 3:** 16bitová znaménková hodnota udává hodnotu nastavující PWM23 do logické 0

**Value Registr 4:** 16bitová znaménková hodnota udává hodnotu nastavující PWM45 do logické 1

**Value Registr 5:** 16bitová znaménková hodnota udává hodnotu nastavující PWM45 do logické 0





Obr. 20: Mechanismus časování PWM

### 1.5.3 Periferie PIT

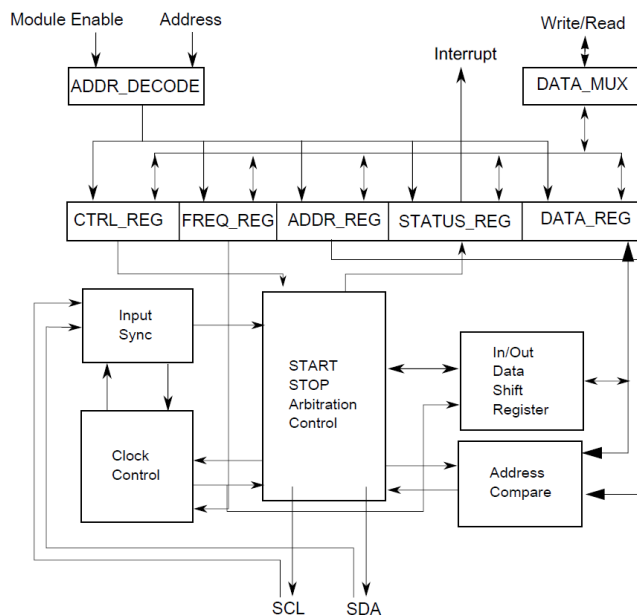
Periodický časovač přerušení (Periodic Interrupt Timer) je periferie, která umožňuje periodické vyvolávání přerušení a triggerování vybraných periférií. Časovač načte počáteční hodnotu z 32bitového *Timer Load Value* registru a tuto hodnotu dekrementuje do 0. Při průchodu 0 se vygeneruje trigger pulz a nastaví se příznak přerušení (interrupt flag). Modul PITu se skládá ze čtyř nezávislých časovačů.

### 1.5.4 Periferie I2C

Mikrokontrolér KV46 disponuje I2C modulem, který mu umožňuje komunikovat s dalšími zařízeními přes I2C sběrnici. Činnost této periferie je řízena pěti registry.

Modul je primárně ovládán pomocí *Control registru*. Vyslání Start a Stop bitu je realizováno zápisem do *Control registru*. Pomocí *Data registru* jsou vysílána/čtena data ze sběrnice. Vyslání a příjem ACK/NACK bitu je realizován hardwarově v samotné periférii.

Vysílání dat na sběrnici je iniciováno zápisem požadovaných dat do 8bitového *Data registru*. Čtení ze sběrnice je iniciováno pomocí dummy read (fiktivní čtení). Čtení následujících bytů ze sběrnice je inicializováno čtením z *Data registru*.



Obr. 21: Blokové schéma I2C periferie [10]

## 1.6 VÝBĚR FUNKČNÍCH BLOKŮ

V následujících podkapitolách budou diskutovány jednotlivé bloky měřicího systému a zdůvodněn jejich výběr. Funkční bloky byly vybírány s ohledem na komunikaci s mikrokontrolérem. Cílem bylo vybrat jednotnou sběrnici pro všechna zařízení – zjednodušení návrhu. Jelikož tyto součástky jsou ve velké míře vyráběny s komunikačním rozhraním I2C a zároveň mikrokontrolér KV46 disponuje I2C periferií, byla zvolena tato sběrnice.

### 1.6.1 Ukládání hodnot

Z funkčního hlediska je vhodné, aby byly naměřené hodnoty teplot uchovávány – zpětná analýza vývoje teploty na jednotlivých částech testovací stolice včetně elektrického motoru. Zaznamenané hodnoty by v případě odpojení napájecího napětí měly zůstat zachovány, což nesplňují paměti typu RAM.

Data budou ukládána v relativně dlouhých časových rámcích v řádech desítek minut a jednotek hodin, s nejmenší periodou vzorkování 1 s. Velikost datového toku ukládaných hodnot nebude velká, hodnota teploty je uložena v proměnné typu *float*, která zabírá 4 byty, což při počtu 5 senzorů činí dohromady 20 bytů. Není tedy kladen důraz na rychlost mechanismu zápisu dat v rámci paměti ani na rychlost komunikace mezi mikrokontrolérem a pamětí.

Z výše uvedených důvodů je nevhodné použít volatilní typ paměti. Vhodnější je použít paměť, ve které data zůstanou i po odpojení napájení (nevolatilní typ paměti). Nejrozšířenějšími pamětmi pro tento druh aplikací jsou EEPROM a FLASH. Vzhledem k malému datovému toku ukládaných dat a možnosti adresace po jednom bytu byla zvolena paměť EEPROM, konkrétně 24LC512 od firmy Microchip [11].

## 1.6.2 RTC

RTC (Real-time clock) je integrovaný obvod, který udržuje hodnotu aktuálního času. Existují v mnoha provedeních, často mívají integrovaný kalendář a možnost externího triggerování po určitých časových úsecích. V této konkrétní aplikaci je RTC modul využitý k přesnému časovému určení naměřené hodnoty teploty.

Konkrétně byl vybrán obvod RTC MCP7940N od firmy Microchip. Obvod obsahuje 64bytovou paměť SRAM, která se skládá z kontrolních a konfiguračních registrů. Výhodou MCP7940N je také zálohování času při odpojení napájecího napětí, což umožňuje externí baterie [13].

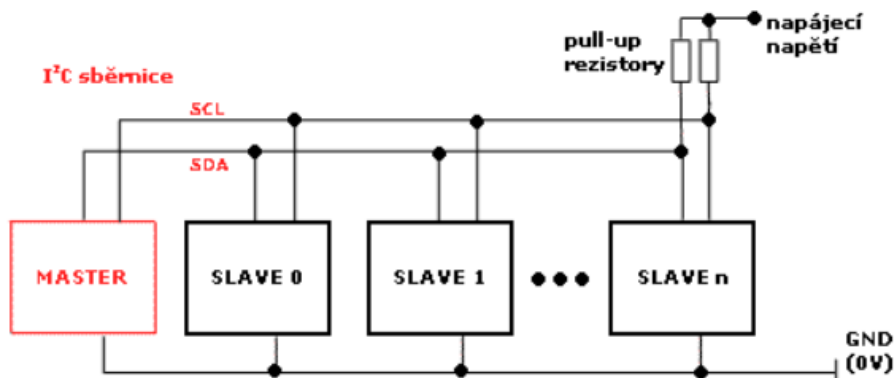
## 1.6.3 Displej

Z důvodu zobrazení měřené teploty a nastavování měřicího systému je zařízení opatřeno zobrazovací jednotkou.

Jelikož mikrokontrolér KV46 pracuje s 3,3 V logikou, tak jedním z hlavních kritérií bylo pracovní napětí displeje 3 V. Dalšími kritérii byla úspora vodičů, snadná komunikace a dostatečně velká zobrazovací plocha (minimálně 4 řádky). Z těchto důvodů byl zvolen 20x4 displej MCOG42005A6W-BNMLWI od firmy Midas komunikující po I2C sběrnici. Displej je ovládaný řadičem SSD1803A [12].

## 1.7 I2C SBĚRNICE

I2C (Internal-Integrated-Circuit Bus) je sběrnice vyvinuta firmou Philips (v současnosti NXP Semiconductors), která slouží nejčastěji k přenosu dat v rámci jednoho zařízení. Uplatňuje se u nízkorychlostních zařízení, například paměť EEPROM. I2C sběrnice umožňuje připojení více master i slave zařízení. Více master zařízení je výhodné například při sdílení externí paměti dvou mikrokontrolérů. Komunikace probíhá po dvou vodičích – SDA (Serial Data) a SCL (Serial Clock). Jak již bylo uvedeno, na sběrnici lze připojit více zařízení. Sběrnice lze zapojit jako multi-master, což znamená, že je zapojeno více zařízení jako master. Jednotlivá zařízení jsou adresována pomocí 7bitové adresy (v rozšířené verzi pomocí 10bitové adresy). [16]



Obr. 22: Schéma I2C sběrnice [16]

### 1.7.1 Komunikace

Zařízení master při přenosu generuje na vodiči SCL hodinový signál. Nejprve musí vyslat adresu zařízení, se kterým chce navázat komunikaci a specifikovat, zda-li půjde o čtení či zápis. To je definováno R/W (Read/Write) bitem, který je součástí adresového bytu.

## 1.8 VÝVOJOVÉ NÁSTROJE

### 1.8.1 Vývojové prostředí

Jako vývojové prostředí je použito MCUXpresso, což je nástroj vyvíjený firmou NXP. Toto prostředí podporuje Arm® Cortex®-M jádra. MCUXpresso rovněž zahrnuje další nástroje, jako například Clock Tool nebo Pin Tool, pomocí kterých lze v grafickém rozhraní nakonfigurovat chování a vlastnosti mikrokontroléru v závislosti na aplikačním použití.

### 1.8.2 Program FreeMASTER

Je softwarový nástroj, který umožňuje debugovat aplikaci v reálném čase, což zahrnuje monitorování proměnných, jejich změnu v reálném čase z uživatelského rozhraní běžícího na straně PC a vizualizaci pomocí časových průběhů monitorovaných proměnných. Program umožňuje tři základní způsoby vizualizace proměnných:

- **Variable Watch:** Vybrané proměnné jsou zobrazeny v tabulce v textovém formátu. Hodnoty proměnných lze měnit přímo v tomto rozhraní.
- **Oscilloscope:** Funkce osciloskopu umožňuje monitorovat proměnné a zobrazovat průběhy v závislosti na čase. Vzorkovací frekvence není garantována a závisí na vytížení mikrokontroléru a komunikační rychlosti rozhraní mezi mikrokontrolérem a uživatelským rozhraním běžícím na PC straně. Tento typ monitorování proměnných je vhodný především pro pomalé děje.
- **Recorder:** Tento typ monitorování proměnných odpovídá funkci digitálního osciloskopu, kde perioda vzorkování interních proměnných je přesně definována. Hodnoty proměnných jsou ukládány do interního bufferu mikrokontroléru a následně pak přeneseny na stranu PC.

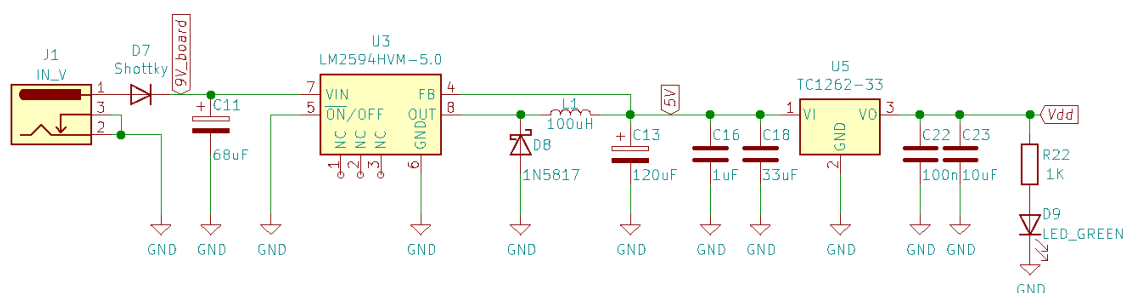
## 2 PRAKTICKÁ ČÁST - HARDWARE

Tato část je zaměřena na návrh a realizaci hardwarové části měřicího systému. Jak již bylo uvedeno v teoretické části, jádrem celého systému je mikrokontrolér KV46 na vývojové desce systému Tower. Konstrukce s mikrokontrolérem umístěným přímo na desce byla zavrhnuta. Tato varianta by výrazně ztížila konstrukci celého zařízení (debugovací rozhraní, mnoho vývodů mikrokontroléru). Vzhledem k těmto skutečnostem a k tomu, že je k dispozici více těchto vývojových desek, byla zvolena varianta propojení vývojové desky s mikrokontrolérem a měřicího systému pomocí hřebenových konektorů J501 a J502 (hřebenové konektory vývojové desky).

### 2.1 NAPÁJECÍ ČÁST

Aby mohlo zařízení fungovat i bez připojení k počítači, bylo nutné navrhnout napájecí část. Vývojová deska s mikrokontrolérem nemá vyvedené napájecí napětí, což je problém, jelikož je potřeba napájet periferie na desce pro měření – displej, RTC, EEPROM atd.

Celé zařízení je tedy napájeno pomocí externího adaptéru s napětím 9 V. Tímto napětím je napájena deska s mikrokontrolérem a spínaný zdroj LM2594HVM, který vytváří napětí 5 V. To slouží jako univerzální 5 V napájecí napětí vyvedené na svorkovnici a zároveň napájí napěťový regulátor TC1262, který vytváří konečné 3,3 V napětí pro napájení měřicí desky. Napěťový regulátor je opatřen dvěma filtračními kondenzátory na vstupu (C16 a C18) a dvěma na výstupu (C22 a C23). LED dioda D9 slouží pouze jako indikace napájecího napětí.

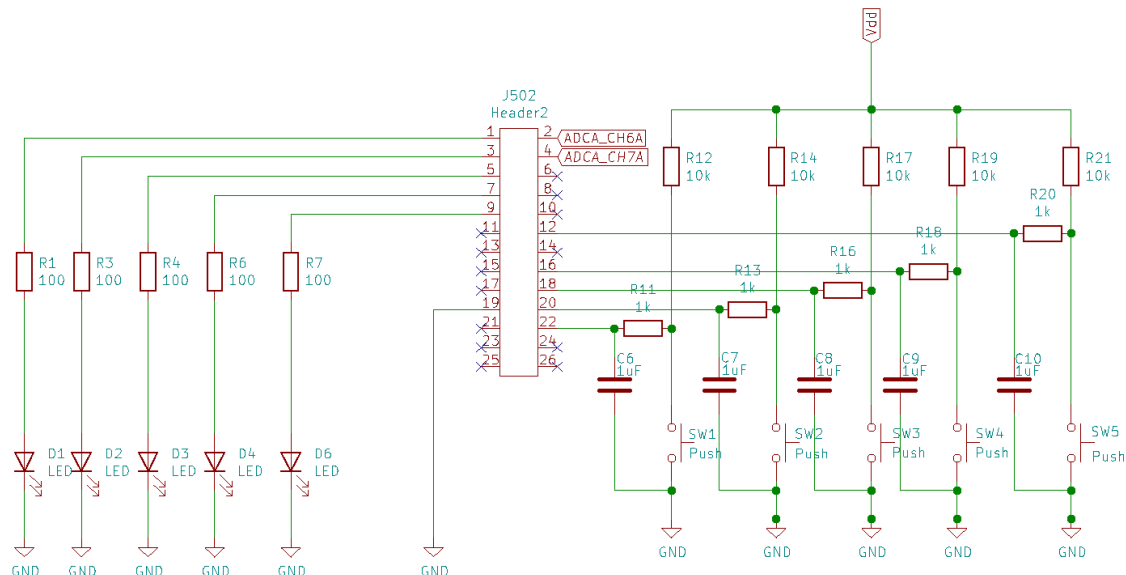


Obr. 23: Zapojení napájecí části

## 2.2 OSTATNÍ ČÁSTI

### Tlačítka a LED diody

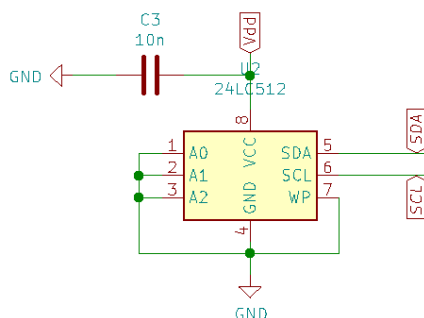
Tato část slouží k ovládání celého zařízení. Tlačítka mají hardwarově ošetřené zákmity pomocí RC členu. LED diody jsou zapojeny k pinům mikrokontroléru v konfiguraci se společnou katodou a slouží jako vizuální signalizace zvýšené teploty senzoru.



Obr. 24: Zapojení ovládací části

### Paměť EEPROM

RTC modul je připojen k I2C sběrnici pomocí pinů *SDA* a *SCL*. Pomocí pinů *A0-A3* je nastavena adresa zařízení pro komunikaci na sběrnici, v tomto konkrétním případě jsou piny připojeny na signál GND. Kondenzátor *C3* slouží jako filtrace vstupního napětí.

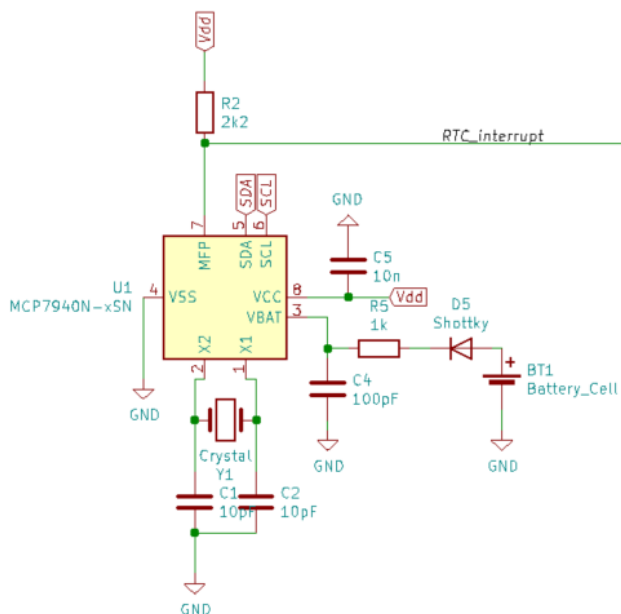


Obr. 25: Zapojení EEPROM paměti

### RTC modul

Stejně jako paměť EEPROM, je tento modul připojen k I2C sběrnici pomocí pinů *SDA* a *SCL*. K pinům *X1* a *X2* je připojen krystal o frekvenci 32,769 kHz. Pin *VBAT* slouží k připojení zálohovacího napětí, v tomto konkrétním případě ploché baterie o napětí 3 V. Pin *MFP* je přes pull-up rezistor připojen k pinu mikrokontroléru a může sloužit

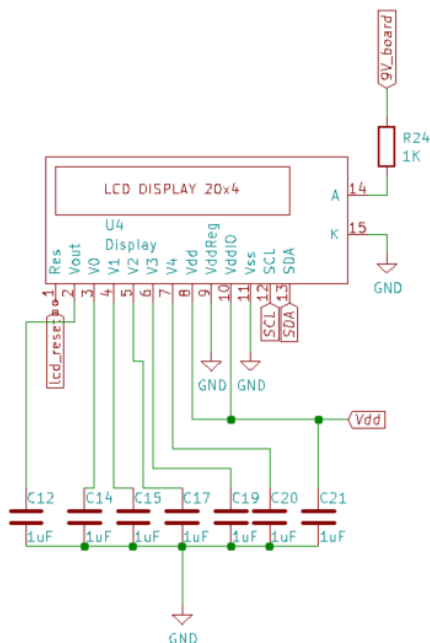
k periodickému vyvolávání přerušení. Kondenzátor C5 slouží jako filtrace vstupního napětí.



Obr. 26: Zapojení RTC modulu

## LCD displej

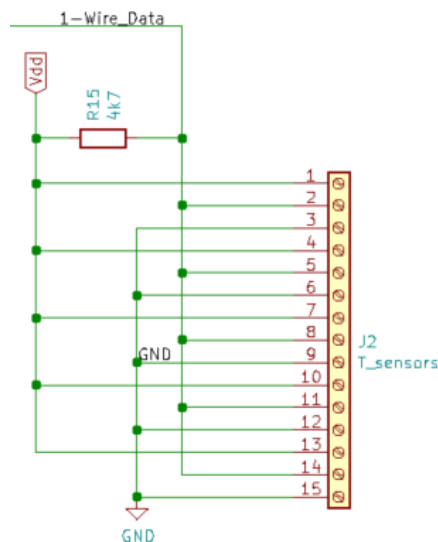
Displej může pracovat s 3 V logikou i s 5 V logikou. Pro každou konfiguraci je nutné displej zapojit jinak podle doporučení v datasheetu [12]. Velmi důležité je připojení pinu *Res* k výstupu mikrokontroléru. Tento pin slouží k resetu řadiče v displeji a bez něj se displej špatně inicializuje. Piny *A* a *K* slouží k připojení napájení pro podsvícení displeje.



Obr. 27: Zapojení LCD displeje

## Teplotní senzory

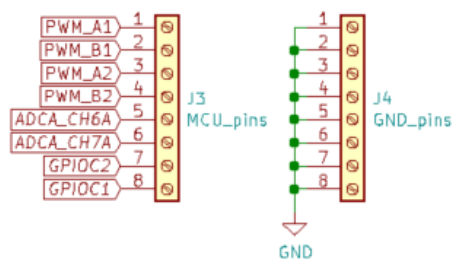
Teplotní senzory jsou připojeny k mikrokontroléru přes dva piny (vstup a výstup pro čtení a zápis z 1-Wire sběrnice). Pro snadnou montáž/demontáž jsou teplotní senzory připojeny přes svorkovnici.



Obr. 28: Zapojení 1-Wire sběrnice

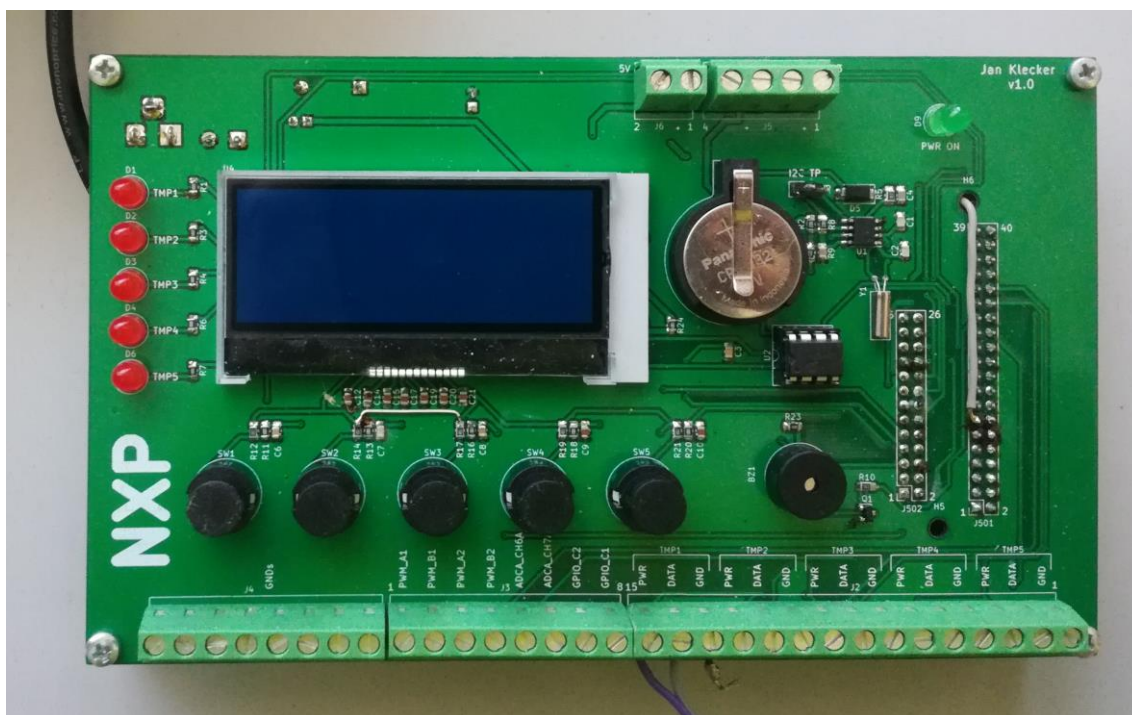
## Univerzální výstupy

Aby mohlo být zařízení více univerzální, jsou na svorkovnice vyvedeny některé piny mikrokonrtoléru.

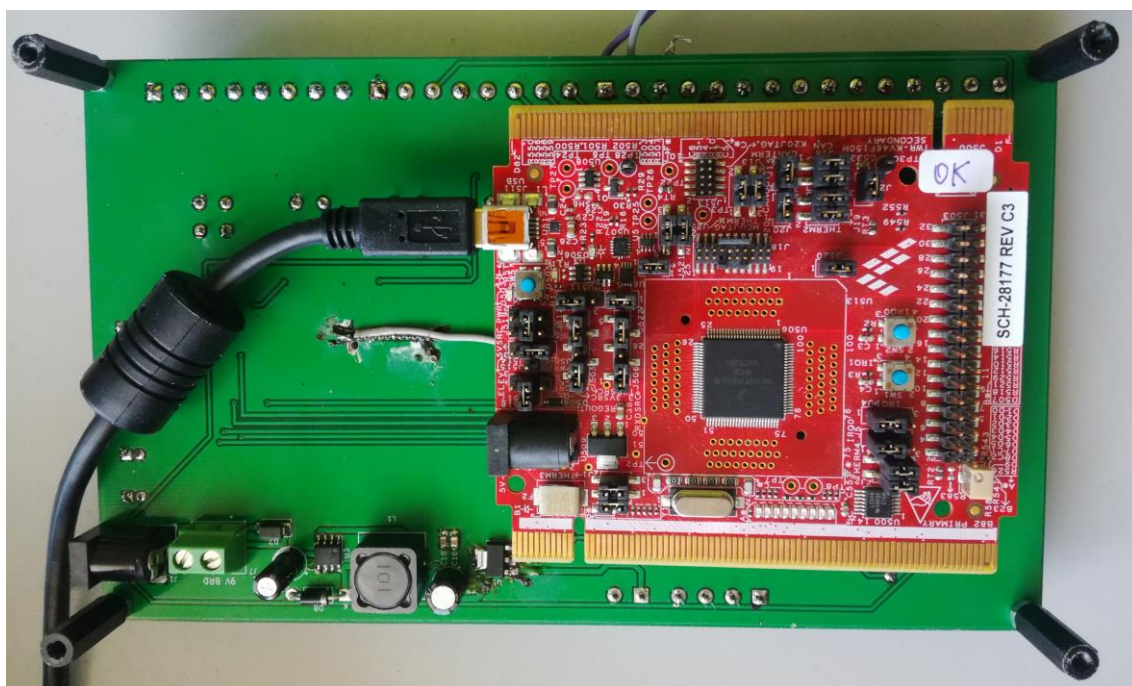


Obr. 29: Zapojení pinů mikrokontroléru





Obr. 30: Osazena DPS, strana TOP

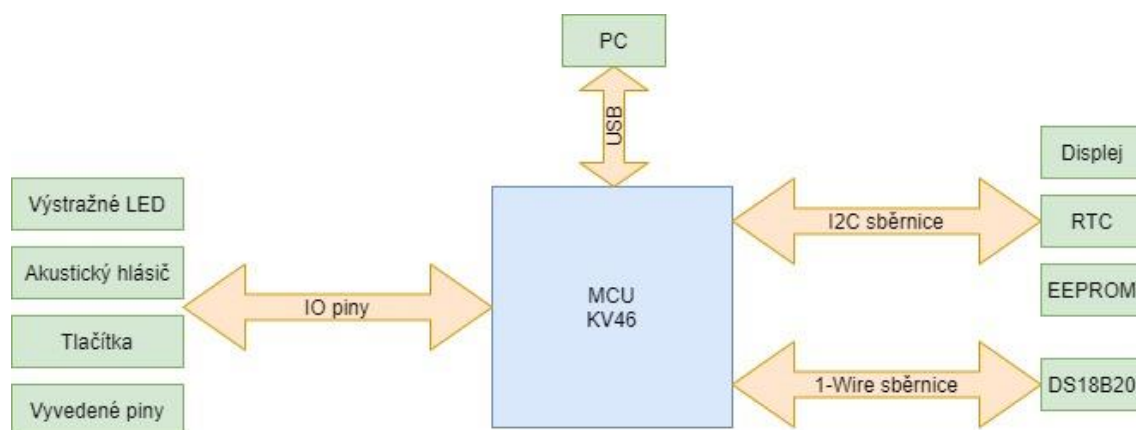


Obr. 31: Osazená DPS, strana BOTTOM

### 3 PRAKTICKÁ ČÁST – SOFTWARE

Cílem návrhu softwaru bylo dodržovat zásady psaní aplikací běžících v reálném čase. Jeden ze základních požadavků je zamezení používání čekacích smyček a důsledné používání návrhu SW na principu stavových automatů. Dalším důležitým požadavkem kladeným na softwarovou implementaci byla možnost snadné integrace algoritmů pro řízení testovací stolice, které mají vyšší prioritu než měření teploty. Ve výsledném návrhu nejsou použity žádné knihovny, veškerý kód je napsán konkrétně pro tento projekt.

Na Obr. 32 je blokové schéma zařízení. Přes I2C sběrnici komunikují displej, RTC modul a EEPROM paměť. Teplotní senzory komunikují přes 1-Wire sběrnici. Ostatní periferie pak komunikují přes GPIO piny. Celé zařízení pak může komunikovat s počítačem přes USB rozhraní.



Obr. 32: Blokové schéma zařízení

Aby byl projekt přehlednější a udržitelný, jsou větší funkční bloky rozděleny do samostatných souborů.

Jelikož je použito mnoho periférií mikrokontroléru, je nutné tyto periferie správně nastavit. K těmto účelům slouží soubor *peripheral\_init.c*, ve kterém jsou deklarované funkce, které inicializují jednotlivé periferie, například *InitFlexPWMA()*. Nastavení hodinového signálu pro jádro a periferie mikrokontroléru bylo vygenerováno pomocí nástroje Clock Tool. Frekvence jádra je nastavena na 95,977472 MHz. Hodinový signál pro celý mikrokontrolér zajišťuje MCG (Multiple Clock Generator). V modulu SIM (System Integrated Module) může být ještě frekvence hodinového signálu upravena pro jednotlivé části mikrokontroléru.

Aby byla celá aplikace přehlednější a lehce pozměnitelná, byly důležité parametry nahrazeny makry, která jsou definovaná v souboru *app\_config.h*. Zásahem v tomto souboru lze tak jednoduše upravit některá důležitá nastavení systému, viz úryvek kódu níže:

```

#define MCU_CLOCK_FREQ          INTERNAL_CLOCK_CORE_CLOCK
#define PWM_FREQ                11111 /* 11.111 kHz */
...
#define TEMP_SENS_1_PLACEMENT   "Stator"
...
#define TEMP_SENS_1_ADDR        0xA000000B9E51F828

```

System je z velké části obsluhován v přerušení. Aby se „kritické“ operace prováděly v přesně daných časových oknech, je nutné ošetřit souběh více přerušení. Přerušování v mikrokontroléru KV46 jsou řízena v modulu NVIC (Nested Vectored Interrupt Controller). Modul podporuje celkem 16 úrovní přerušování – čím nižší úroveň, tím vyšší priorita přerušování. Proto jsou součástí souboru *app\_config.h* makra, která definují úrovně přerušování pro jednotlivé periferie:

```

#define FLEX_PWM_RELOAD_ISR_PRIORITY    2
#define FLEX_PWM_COMPARE_ISR_PRIORITY  1

```

### 3.1 BEZPEČNÉ MAZÁNÍ BITŮ

Bit	15	14	13	12	11	10	9	8
Read	0	RUF	REF	RF	CFA1	CFA0	CFB1	CFB0
Write			w1c	w1c	w1c	w1c	w1c	w1c
Reset	0	0	0	0	0	0	0	0

Bit	7	6	5	4	3	2	1	0
Read	CFX1	CFX0	CMPF					
Write	w1c	w1c	w1c					
Reset	0	0	0	0	0	0	0	0

Obr. 33: Status registr periferie PWM [10]

V registrech, ve kterých se vyskytují status flagy (asynchronně nastavované bity, které informují o stavu určité události) zároveň s konfiguračními bity, je nutné zvolit vhodný mechanismus, který bude bezpečně mazat status flagy. Příklad bude uveden na periférii PWM, u které je nutné tyto okolnosti ošetřit. Použití následující konstrukce je špatné. Bit *RF* se sice vymaže, ale jestliže je jakýkoliv jiný bit přečten jako 1, při opětovném zápisu do registru se daný bit vymaže, což vede k hazardnímu stavu, který se jen velmi těžko debuguje v reálném čase.

```

#define PWM_STS_RF_MASK          0x1000 // Reload Flag
PWMA->SM[0].STS |= PWM_STS_RF_MASK;

```

Je tedy nutné ošetřit zápis tak, aby se při zápisu ovlivňoval jen požadovaný bit. Toho docílíme vhodnou obecnou konstrukcí, která zabezpečí zápis logické 1 na pozici mazaného flagu a na pozice zbývajících flagů je bez ohledu na jejich aktuálním stavu zapsaná logická 0. Je zde nutno podotknout, že mechanismus mazání flagů probíhá zápisem logické 1 pokud je flag nastaven, pak dojde k jeho automatickému smazání. Pokud je flag nastaven a dojde k zápisu logické 0 na danou pozici, flag se nesmaže a jeho stav je tudíž nezměněn. Ukázka makra, které slouží k bezpečnému mazání bitů:

```
#define PWM_CLEAR_FLAG(param, sm) PWMA->SM[sm].STS &= ~(~(param) \
& (EFPWM_STS_REF | EFPWM_STS_RF | EFPWM_STS_CAPT_MASK | \
EFPWM_STS_CMPF_MASK));
```

## 3.2 NÁVRH SOFTWARE PRO 1-WIRE SBĚRNICI

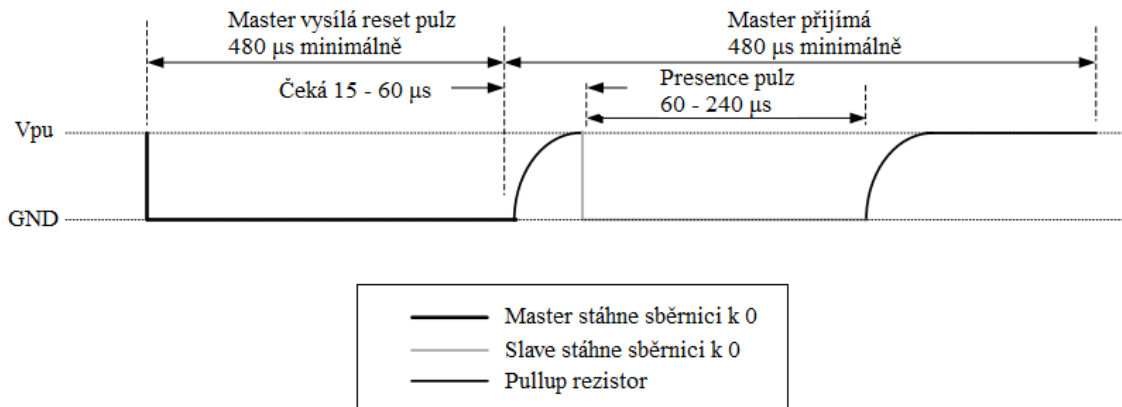
Následující podkapitola je zaměřena na návrh softwaru pro ovládání teplotních senzorů DS18B20 na 1-Wire sběrnici.

### 3.2.1 Nevhodnost oficiální knihovny

Při hledání způsobů ovládání 1-Wire sběrnice bylo prozkoumáno mnoho zdrojů, drtivá většina z nich však obsahovala mechanismus, který je pro tuto aplikaci naprosto nevhodný. Příklad bude popsán na kódu z aplikační noty senzoru DS18B20 z oficiálních webových stránek firmy Maxim Integrated.

V ukázce kódu níže [14] je funkce, která slouží k inicializaci sběrnice a následnému čtení presence pulzu. Hlavním problémem této funkce, potažmo celé knihovny, jsou *delay()* funkce.

```
unsigned char ow_reset(void)
{
    unsigned char presence;
    DQ = 0;                // pull DQ line low
    delay(29);             // leave it low for 480us
    DQ = 1;                // allow line to return high
    delay(3);              // wait for presence
    presence = DQ;         // get presence signal
    delay(25);             // wait for end of timeslot
    return(presence);      // presence signal returned
}                          // 0=presence, 1 = no part
```



Obr. 34: Průběh reset pulzu 1-Wire sběrnice [12]

Funkce `delay(int useconds)` v tomto kódu zajišťuje potřebné časy logických úrovní na sběrnici. Ze znalosti taktu hodinového signálu mikrokontroléru je funkci předán parametr, na kterém pak závisí doba čekání. Definice této funkce [12] vypadá následovně:

```
void delay(int useconds)
{
    int s;
    for(s=0; s<useconds;s++);
}
```

Blok `sfor()` cyklem pak plní funkci čítače, který inkrementuje proměnnou `s` do doby, než je rovna parametru `userconds`. Jde tedy o takzvanou čekací smyčku. Mikrokontrolér po dobu čekací smyčky nevykonává žádný užitečný kód a je tedy jeho výpočetní výkon nevyužitý. Při komplexnější operaci, kdy funkce obsahuje více funkcí `delay()`, jako je například čtení ze sběrnice, je mikrokontrolér zahlcený po dobu trvání této funkce. Jak již bylo zmíněno výše, monitorovací systém teploty bude pouze část celku obsahující například řízení výkonu motoru, což znamená okamžité reakce na vnější podněty. Z tohoto důvodu je použití této konstrukce nevhodné.

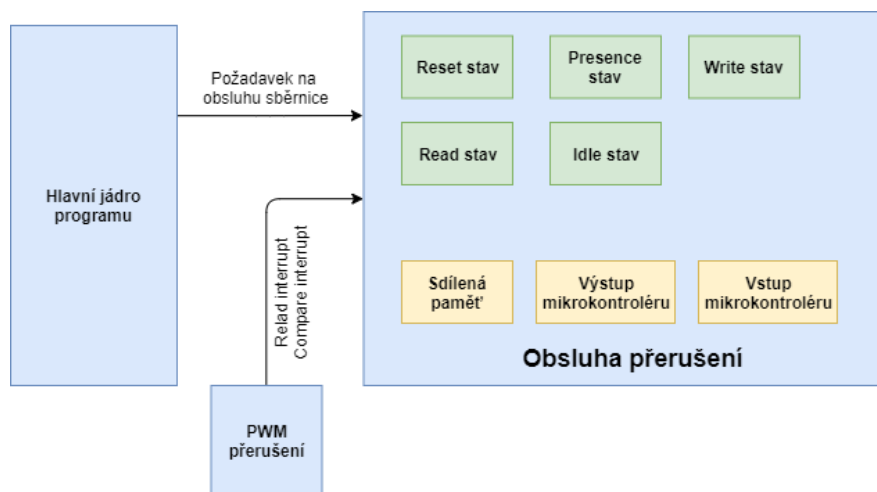
### 3.2.2 Řízení sběrnice periférií PWM

Jako náhradu za čekací smyčky je vhodné použít systém přerušení. Přerušení, anglicky `interrupt`, je metoda pro asynchronní obsluhu událostí, kdy procesor přeruší vykonávání sledu instrukcí, vykoná obsluhu přerušení, a pak pokračuje v předchozí činnosti. Požadavky na přerušení obecně přichází z periférií, jádra mikrokontroléru nebo mohou být i softwarové. Výhoda tohoto časování spočívá v tom, že časování, která by normálně běžela v rámci hlavní části programu, mohou být realizována v systému přerušení. U mikrokontroléru MKV46F256VLL16 byla zvolena pro tyto účely periferie `eFlexPWM`.

Řešení pomocí PWM je výhodné z několika důvodů. Časování `time slotů` (jak čtecích, tak zápisových) je řešené snadno a přehledně pomocí *Value registrů*. Další

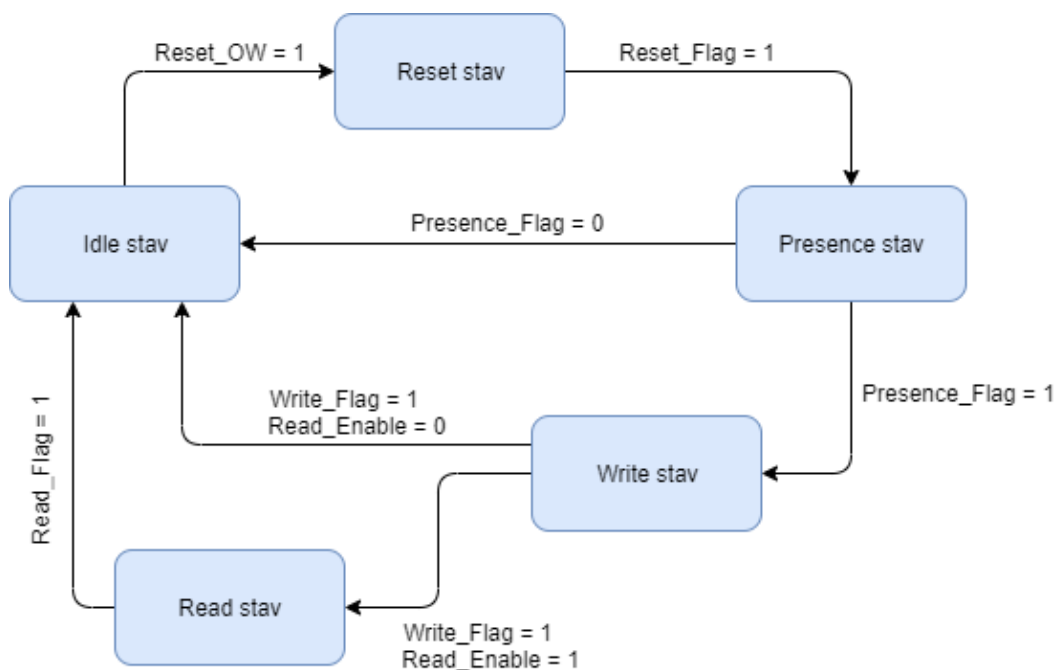


výhodou je vyvedení výstupu PWM přímo na sběrnici, tudíž odpadá nutnost řídit výstup mikrokontroléru pomocí softwaru. Navíc všechny tyto úkony jsou řešeny v rámci přerušení.



Obr. 35: Princip řízení 1-Wire sběrnice

Program na obsluhu sběrnice je navržen pomocí stavového automatu s pěti stavy. Při zápisu na sběrnici je buffer naplněn příkazy a popřípadě hodnotami, a následně je celý zapsán na sběrnici. Odpadá tedy nutnost hlídat si jednotlivé příkazy a hodnoty, pouze stačí naplnit buffer a celé pole se zapíše na sběrnici.



Obr. 36: Stavový automat ovládání sběrnice

**Reset stav:** V tomto stavu je na sběrnici vyslán reset pulz.

**Presence stav:** V tomto stavu je sběrnice čtena. Jestliže se na sběrnici vyskytuje alespoň jedno zařízení, tak vyvolá presence pulz. V případě, že master indikuje tento

pulz, pokračuje dále, jinak se vrací do Idle stavu.

**Write stav:** V tomto stavu dochází k zápisu. Buffer je naplněn příkazem a popřípadě hodnotami, a je vyslán na sběrnici. Jednotlivé bity jsou načítány vždy při *Reload interruptu*.

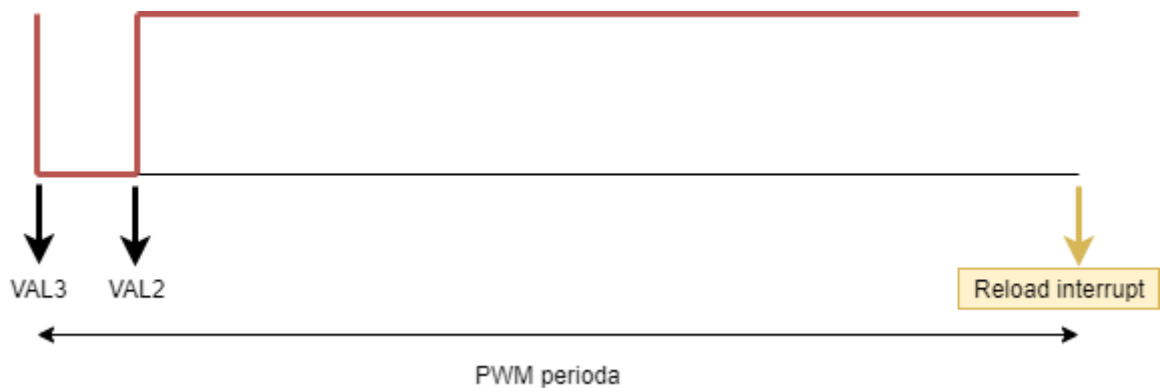
**Read stav:** Příkazy ke čtení využívají Read state. Do tohoto stavu se program dostává při povolení čtení, příznak *Read\_Enable*. Jednotlivé bity jsou čteny v *Compare interruptu*.

### Zápis na sběrnici:

Zápis probíhá v časových slotech. Trvání časového slotu nastavíme pomocí PWM periody, tedy pomocí *Init registru* a *Value registru 1*. Pomocí *Value registru 2 a 3* jsou nastavené časy logické 0 a logické 1 sběrnice. Stejným principem je vysílán i reset pulz.

Při ukončení jednoho PWM cyklu dojde k načtení bufferovaných hodnot *Value registrů* a dojde k *Reload interruptu*. V obsluze tohoto přerušení je nastavena střída (odpovídající logické 0 nebo 1), která je načtena při vyvolání dalšího *Reload interruptu*. Inicializace zápisu logické 1 pomocí *Value registrů* vypadá následovně:

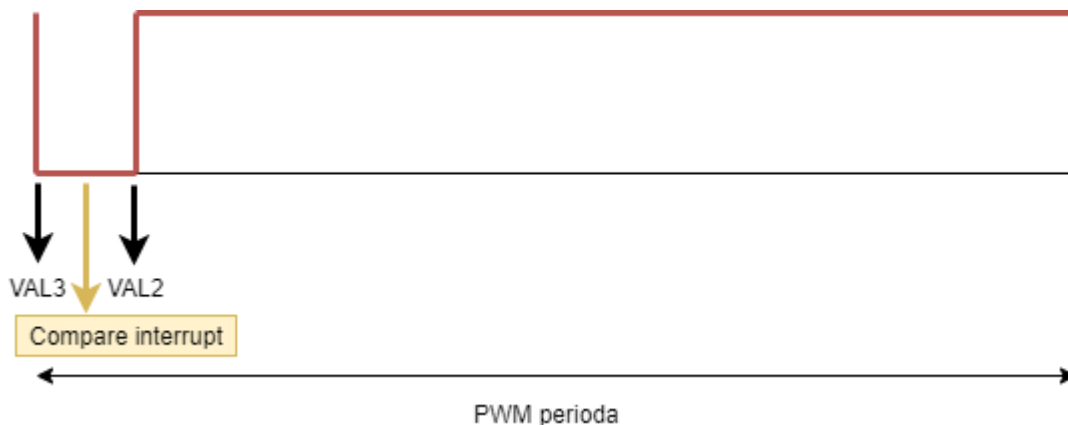
```
/* Write 1 time slot */
PWMA->SM[0].VAL3 = (uint16_t)(0);
PWMA->SM[0].VAL2 = (uint16_t)(150);
PWMA->SM[0].VAL1 = (uint16_t)(PWM_MODULO);
PWMA->MCTRL |= PWM_MCTRL_LDOK(15);
```



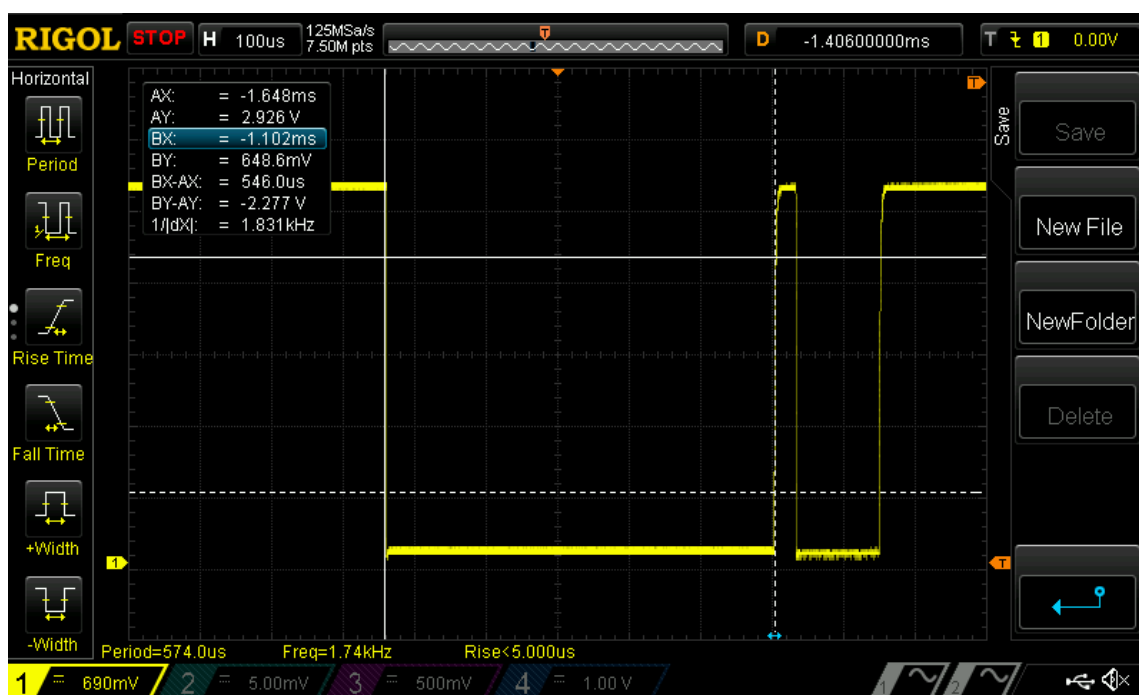
Obr. 37: Zápis logické 1 na sběrnici

### Čtení ze sběrnice:

Pro čtení ze sběrnice nejprve musí mikrokontrolér sběrnici stáhnout k logické 0 a poté ji uvolnit. Slave na toto reaguje tím, že buď sběrnici podrží v logické 0 (slave vysílá logickou 0) nebo ponechá sběrnici uvolněnou (slave vysílá logickou 1). Časový slot (PWM perioda) zůstává stejný jako u zápisu. Master pak musí číst sběrnici v daném časovém rozmezí. Toto časování je zajištěno pomocí *Compare interruptu*, který se vyvolá, když hodnota časovače PWM je stejná jako hodnota *Value registru*. V obsluze tohoto přerušení je realizováno čtení ze sběrnice.



Obr. 38: Čtení ze sběrnice



Obr. 39: Průběh na sběrnici – reset (vlevo) a presence (vpravo) pulz

### 3.3 NÁVRH SOFTWARE PRO I2C SBĚRNICI

Jak už bylo zmíněno v teoretickém úvodu, mikrokontrolér KV46 disponuje I2C periferií. Při zápisu/čtení sběrnice probíhá přenos dat po bytech, které jsou zakončeny ACK/NACK bitem. K přenosu jednoho bytu je tedy potřeba 9 tiků hodinového signálu. Zahájení/ukončení komunikace se v I2C periferii vyvolává nastavením/mazáním bitu MST v *Control registru 1*, což zajišťuje vyslání *Start bitu* nebo *Stop bitu*. Při přenosu je tedy nutné vyslat Start bit a poté zapsat první byte do *Data I/O registru*, čímž se vyvolá jeho přenos na sběrnici. Pro přenos dalšího bytu je tedy nutné počkat, než se odešle ten předchozí. Aby odpadla potřeba softwarově kontrolovat, zda-li byl byte již opravdu poslán, byla navržena konstrukce se systémem přerušení.



Po odeslání/přijetí bytu je v *Status registru* nastaven bit *TCF* do 1, čímž je vyvoláno přerušení. V přerušení je zápisem dalšího bytu do *Data I/O registru* vyvolán další přenos. Celá konstrukce tedy spočívá v tom, že je pole bytů naplněno adresou I2C zařízení a daty. Při vyvolání přerušení se inkrementuje ukazatel na položku v zápisovém poli, čímž dojde k vyslání dalšího bytu. K řízení celého procesu slouží struktura *I2CStruct*.

***slave\_address***: Obsahuje I2C adresu zařízení, se kterým je právě komunikováno.

***read\_enable***: Příznak, jestli po zápisu následuje čtení.

***write\_byte\_cnt***: Udává počet bytů, které mají být vyslány z *write\_bufferu*.

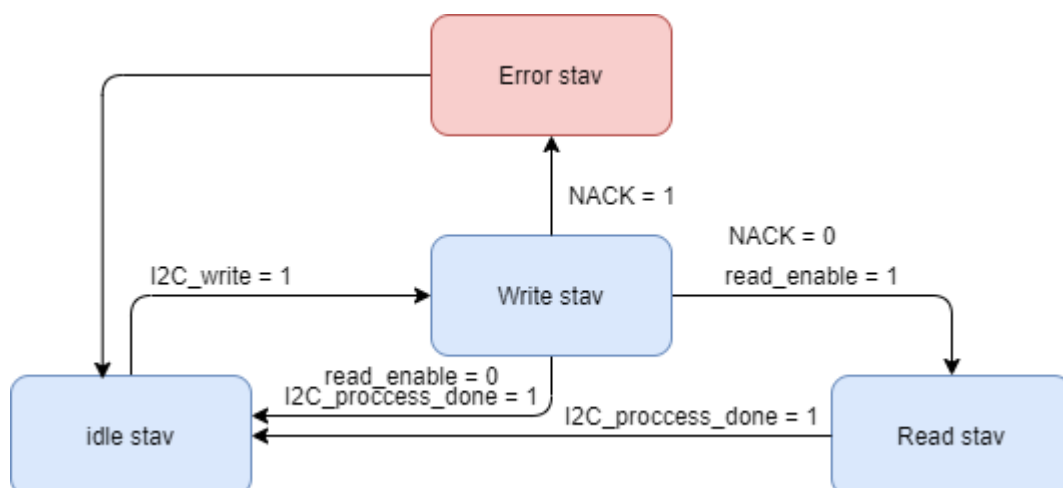
***read\_byte\_cnt***: Udává počet bytů, které mají být přijaty.

***read\_buffer***: Při čtení ze sběrnice se do tohoto pole ukládají přijaté byty.

***write\_buffer***: Obsahuje byty, které mají být zapsány.

```
/* I2C Structure */
typedef struct
{
    Uint8_t slave_address;
    Uint8_t read_enable;
    Uint8_t write_byte_cnt;
    Uint8_t read_byte_cnt;
    Uint8_t read_buffer[20];
    Uint8_t write_buffer[83];
} I2CStruct;
```

Obsluha sběrnice je pak řízena stavovým automatem, který je na Obr. 40.



**Obr. 40: Stavový automat pro I2C sběrnici**

**Idle stav:** Stav, ve kterém nedochází ke komunikaci se sběrnici.

**Write stav:** Stav, ve kterém dochází k zápisu na sběrnici. Jestliže neobdrží potvrzovací bit, vrací se zpátky do Idle stavu přes Error stav.

**Read stav:** Stav, ve kterém dochází ke čtení ze sběrnice.

**Error stav:** Stav, do kterého přechází z Write stavu, když neobdrží potvrzovací bit. Tento stav slouží pouze k indikaci problému, poté se automat vrátí do Idle stavu.

## 3.4 NÁVRH MĚŘICÍHO SOFTWARE

Tato část návrhu je zaměřena na návrh softwaru obsluhující měření a zpracování teploty. Z hlediska využitelnosti byly navrženy dva režimy měření teploty. V obou režimech dochází k měření teploty všemi připojenými senzory a je možnost nastavit parametry senzorů - hodnotu teploty alarmu a akustické hlášení překročené teploty. Nejmenší perioda měření teploty je 1 sekunda.

### Režim s ukládáním teploty

První režim s ukládáním teploty se hodí pro zpětnou analýzu vývoje teploty na elektrickém motoru a kritických částech testovací stolice. Při inicializaci měření je nejprve vymazána paměť EEPROM, poté je načten čas (začátek měření) z RTC modulu a s periodou měření je uložen do paměti. Následně je spuštěno samotné měření teploty. Časové trvání testu a perioda ukládání hodnot lze nastavit.

Teploty ze senzorů jsou ukládány do 4bytové proměnné typu *float*. Při počtu pěti senzorů tedy každou periodu musí dojít k uložení 20 bytů do paměti EEPROM. Při zápisu do paměti je vždy poslána vnitřní adresa paměti následovaná jednotlivými byty. Paměť disponuje vnitřním ukazatelem na adresu, který se po zápisu každého bytu sám inkrementuje. Po překročení adresy aktuální stránky (page) ukazatel přeteče na adresu prvního bytu stránky. Při ukládání je tedy nutné hlídat si v proměnné aktuální adresu v paměti a v případě, že se právě ukládaná data nevejdou na danou stránku, tak zápis začít na stránce nové.

Jak již bylo uvedeno výše, teploty jsou v mikrokontroléru uloženy v proměnné typu *float*. Aby bylo možné teplotu uložit, musí být „rozkouskována“ na jednotlivé byty. Bohužel, nad proměnnými s plovoucí desetinnou čárkou nelze provádět binární operace, není tedy možné vymaskovat jednotlivé byty. Proto byla zvolena konstrukce pomocí datového typu *union*:

```
/* Temperature union */
union Temperature
{
    float SensorTemperature;
    uint_8t TemperatureArr[4];
}TemperatureUnion;
```

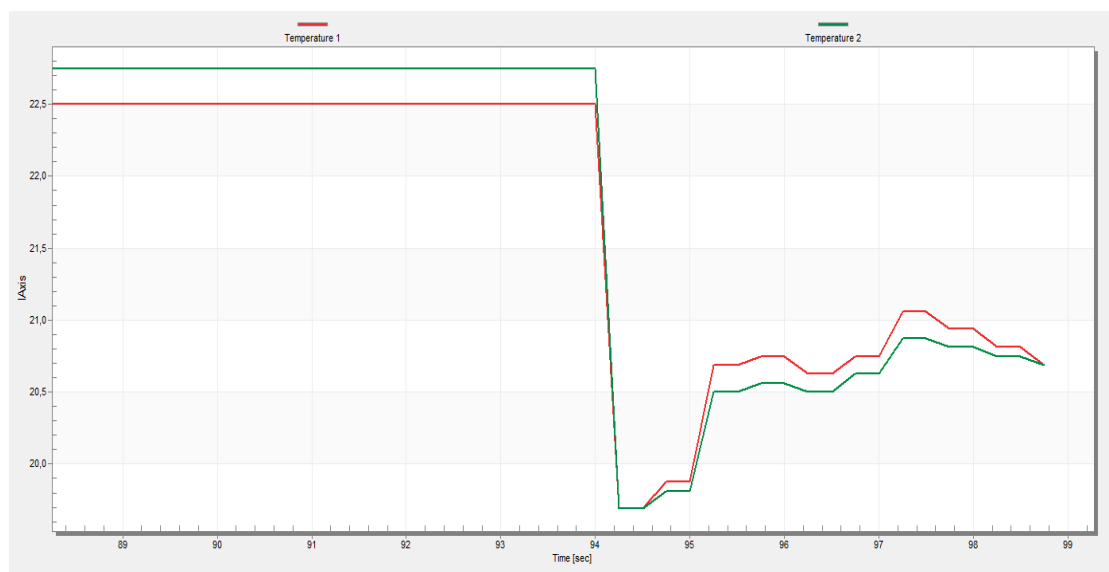
*Union* v paměti zabere velikost největší proměnné uložené v *unionu*. Do proměnné typu *float* je uložena teplota. Jelikož se obě proměnné překrývají, je možné pomocí pole *TemperatureArr* získat jednotlivé byty proměnné *float*. Ukládaná data je možné následně zpracovat pomocí programu FreeMaster.

### Režim bez ukládání teploty

Ve druhém režimu dochází pouze ke snímání teploty s pevnou periodou 1 sekunda.

### Režim vizualizace dat

V tomto režimu je umožněna zpětná vizualizace dat pomocí programu FreeMaster. Obsah paměti EEPROM je periodicky vyčítán a data jsou zobrazována v osciloskopu FreeMasteru.



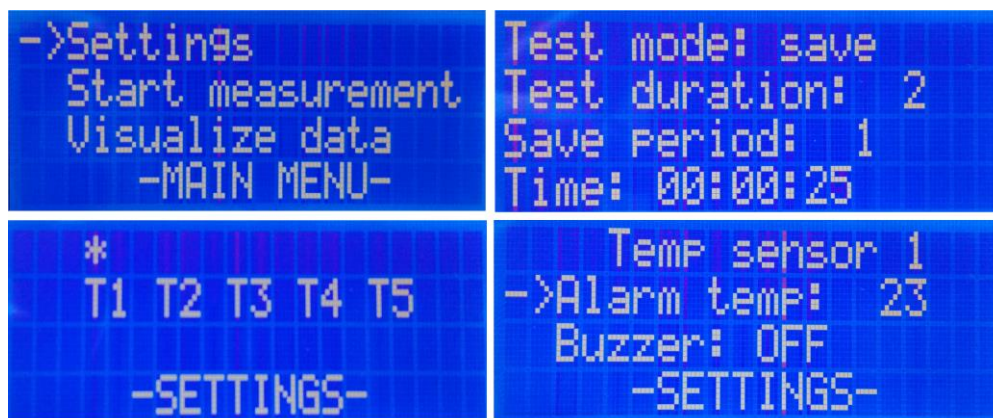
Obr. 41: Průběh teploty uložené v paměti EEPROM v programu FreeMaster

## 3.5 UŽIVATELSKÉ ROZHRAŇÍ

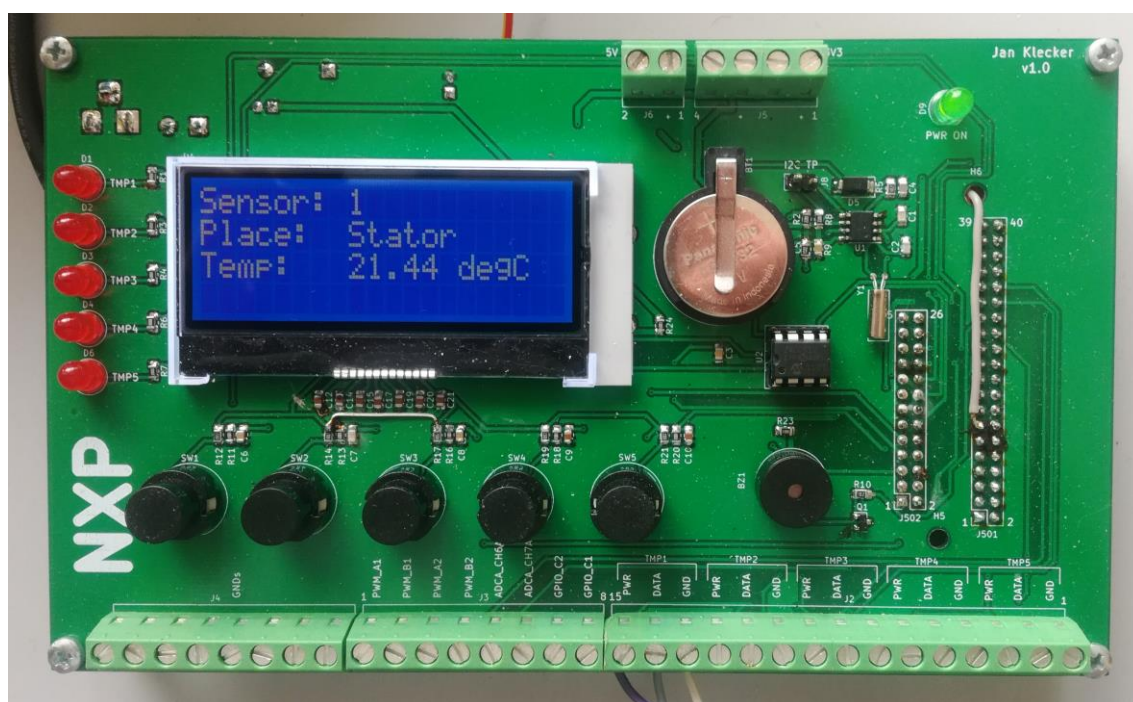
Pro zobrazování dat a ovládání zařízení bylo navrženo uživatelské rozhraní ovládané pomocí tlačítek. Celý systém je pak navržen jako stavový automat, mezi jehož stavy se přepíná tlačítky. Dělí se na funkční část a nastavovací část.

Ve funkční části dochází k samotnému měření, popřípadě k vizualizaci dat. Při měření teploty jsou na základní obrazovce vypsány informace o právě probíhajícím testu. Pomocí tlačítek pak lze přepínat mezi jednotlivými teplotními senzory.

V nastavovací části může uživatel nastavit některé parametry zařízení – druh testu, jeho trvání (hodiny) a periodu ukládání (sekundy).



Obr. 42: Ukázka uživatelského rozhraní



Obr. 42: Ukázka funkčnosti zařízení

## 4 ZÁVĚR

Cílem této bakalářské práce bylo seznámit se s problematikou měření teploty, s mikrokontroléry firmy NXP Semiconductors, s návrhem plošných spojů a s komplexní realizací měřicího systému pro testovací stolicí firmy NXP Semiconductors. Zařízení mělo být postaveno na vhodném mikrokontroléru s přihlédnutím na možná rozšíření. Dále bylo úkolem navrhnout ovládací software pro celé zařízení, který by příliš nezatěžoval výpočetní výkon mikrokontroléru.

V teoretické části byly diskutovány rozdíly mezi analogovými a digitálními teplotními senzory. Z hlediska odolnosti proti šumu a možnosti vedení senzoru na delší vzdálenost byl vybrán senzor DS18B20. V požadavcích zadání práce bylo použití vhodného mikrokontroléru od firmy NXP Semiconductors. Vzhledem k tomu, že systém má být univerzální a rozšířený o bloky řízení motoru, byl zvolen mikrokontrolér MKV46F256VLL16 na vývojové desce Tower, který je k těmto účelům používán.

V kapitole Návrh hardwaru byly rozebrány jednotlivé části měřicí desky a ukázán její návrh. Deska pro měření byla s vývojovou deskou Tower spojena pomocí hřebenových konektorů. Pro připojení periférií byla vybrána jednotná sběrnice I2C, jedním z hlavních důvodů byla hardwarová periferie I2C v mikrokontroléru KV46. Do svorkovnice na měřicí desce jsou vyvedeny některé další piny mikrokontroléru – ADC, PWM.

V kapitole Návrh softwaru byl navržen kód pro ovládání I2C a 1-Wire sběrnice. Ovládání senzorů připojených na 1-Wire sběrnici bylo vyřešeno elegantně pomocí periferie PWM, kdy zápis logické 0 či 1 je řízen její střídou. Pro měření teploty byly navrženy 2 módy, s ukládáním teploty a bez ukládání teploty. Teplota je ukládána do paměti EEPROM. Průběh teploty z jednotlivých senzorů uložených v paměti EEPROM lze pomocí osciloskopu v programu FreeMaster vizualizovat. Ke snadnému ovládání a nastavování měření bylo navrženo uživatelské rozhraní.

Bylo tedy navrženo funkční zařízení s pěti senzory teploty, které umožňuje monitorovat teplotu kritických částí testovací stolice a elektrických motorů v průběhu vývoje řídicích algoritmů moderních elektrických pohonů.

# LITERATURA

- [1] *Different Types of Sensors – Analog and Digital* [online]. ©2017. [Cit. 27.11.2019]. Dostupné z: <https://www.codrey.com/electronics/different-types-of-sensors/>
- [2] PAONESSA, Simon. *Reducing Signal Noise in Practice* [online]. ©2017. [Cit. 29.11.2019]. Dostupné z: [https://www.predig.com/sites/default/files/documents/whitepapers/Reducing\\_Signal\\_Noise.pdf](https://www.predig.com/sites/default/files/documents/whitepapers/Reducing_Signal_Noise.pdf)
- [3] *The Operating Principle of Thermocouples* [online]. ©2019. [Cit. 2.12.2019]. Dostupné z: <https://blog.belilove.com/2019/06/the-operating-principle-of-thermocouples.html>
- [4] SEITZ, Jason. *Designing Thermocouple Applications with a sensor AFE* [online]. ©2011. [Cit. 15.11.2019]. Dostupné z: <https://www.digikey.be/nl/articles/techzone/2011/may/designing-thermocouple-applications-with-a-sensor-afe>
- [5] *Temperature Sensors – Types, Working & Operation* [online]. ©2011. [Cit. 16.11.2019]. Dostupné z: <https://www.elprocus.com/temperature-sensors-types-working-operation/>
- [6] *Thermally Sensitive Resistors* [online]. [Cit. 19.4.2020]. Dostupné z: <https://www.peakensors.co.uk/resources/thermistor-information/>
- [7] DAVIS, Nick. *Introduction to Temperature Sensors: Thermistors, Thermocouples, RTDs, and Thermometer ICs* [online]. ©2011. [Cit. 27.2.2020]. Dostupné z: <https://www.allaboutcircuits.com/technical-articles/introduction-temperature-sensors-thermistors-thermocouples-thermometer-ic/>
- [8] *Maxim Integrated. Programmable Resolution 1-Wire Digital Thermometer DS18B20* [online]. [cit. 13. 10. 2019] Dostupné z: <https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>
- [9] *TWR-KV46F150M: Kinetis KV4x Family Tower System Module* [online]. [cit. 19. 3. 2020] Dostupné z: <https://www.nxp.com/design/development-boards/tower-development-boards/mcu-and-processor-modules/kinetis-modules/kinetis-kv4x-family-tower-system-module:TWR-KV46F150M>
- [10] *KV4x Reference Manual* [online]. ©2011. [cit. 16. 3. 2020] Dostupné z: <https://www.nxp.com/design/development-boards/tower-development-boards/mcu-and-processor-modules/kinetis-modules/kinetis-kv4x-family-tower-system-module:TWR-KV46F150M>
- [11] *512K I2C™ Serial EEPROM* [online]. Dostupné z: <http://ww1.microchip.com/downloads/en/devicedoc/21754m.pdf>
- [12] *MCCOG42005A6W-BNMLW Datasheet* [online]. Dostupné z: <https://datasheet.octopart.com/MCCOG42005A6W-BNMLWI-Midas-datasheet-62095894.pdf>
- [13] *MCP7940N Datasheet* [online]. Dostupné z: <http://ww1.microchip.com/downloads/en/DeviceDoc/MCP7940N-Battery-Backed-I2C-RTCC-with-SRAM-20005010G.pdf>
- [14] *INTERFACING THE DS18X20/DS1822 1-WIRE® TEMPERATURE SENSOR IN A MICROCONTROLLER ENVIRONMENT* [online]. [Cit. 14.4.2020]. Dostupné z: <https://www.maximintegrated.com/en/design/technical-documents/app-notes/1/162.html>

- [15] *KV4x Data Sheet* [online]. ©2011. [cit. 20. 3. 2020] Dostupné z: <https://www.nxp.com/docs/en/data-sheet/KV4XP100M168.pdf>
- [16] *Stručný popis sběrnice I2C a její praktické využití k připojení externí eeprom 24LC256 k mikrokontroléru PIC16F877* [online]. [cit. 17. 4. 2020] Dostupné z: <https://vyvoj.hw.cz/navrh-obvodu/strucny-popis-sbernice-i2c-a-jeji-practicke-vyuziti-k-pripojeni-externi-eeeprom-24lc256>

## SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

NTC	Negative temperature coefficient, negativní teplotní koeficient
PTC	Positive temperature coefficient, pozitivní teplotní koeficient
IO	Integrovaný obvod
ADC	Analog-to-digital converter, analogově digitální převodník
SRAM	Static random-access memory, statická paměť
EEPROM	Electrically Erasable Programmable Read-Only Memory, elektricky vymazatelná paměť pouze pro čtení
ROM	Read-only memory
CRC	Cyclic redundancy check, cyklický redundanční součet
PWM	Pulse width modulation, pulzní šířková modulace
PIT	Periodic interrupt timer, periodický časovač přerušení
MCU	Microcontroller, mikrokontrolér
LQFP	Low-profile Quad Flat Package

## SEZNAM OBRÁZKŮ

<b>Obr. 1: Testovací stolice s elektromotorem</b> .....	2
<b>Obr. 2: Blokové schéma analogového senzoru [1]</b> .....	3
<b>Obr. 3: Blokové schéma digitálního senzoru [1]</b> .....	3
<b>Obr. 4: Šum v analogovém a digitálním signálu [2]</b> .....	4
<b>Obr. 5: Princip termočlánku [3]</b> .....	4
<b>Obr. 6: Závislost generovaného napětí na teplotě pro různé druhy termočlánků [4]</b> .....	5
<b>Obr. 7: Závislost odporu na teplotě pro PTC a NTC [6]</b> .....	5
<b>Obr. 8: Závislost výstupního napětí na teplotě senzorů firmy Analog Devices [7]...</b>	6
<b>Obr. 9: Rozložení pinů [8]</b> .....	8
<b>Obr. 10: Vnitřní uspořádání senzoru DS18B20 [8]</b> .....	9
<b>Obr. 11: Registr pro ukládání teploty [8]</b> .....	9
<b>Obr. 12: Uspořádání 1-Wire sběrnice</b> .....	11
<b>Obr. 13: Časový průběh reset pulzu [8]</b> .....	12
<b>Obr. 14: Time sloty pro zápis log 0. a log. 1 [8]</b> .....	12



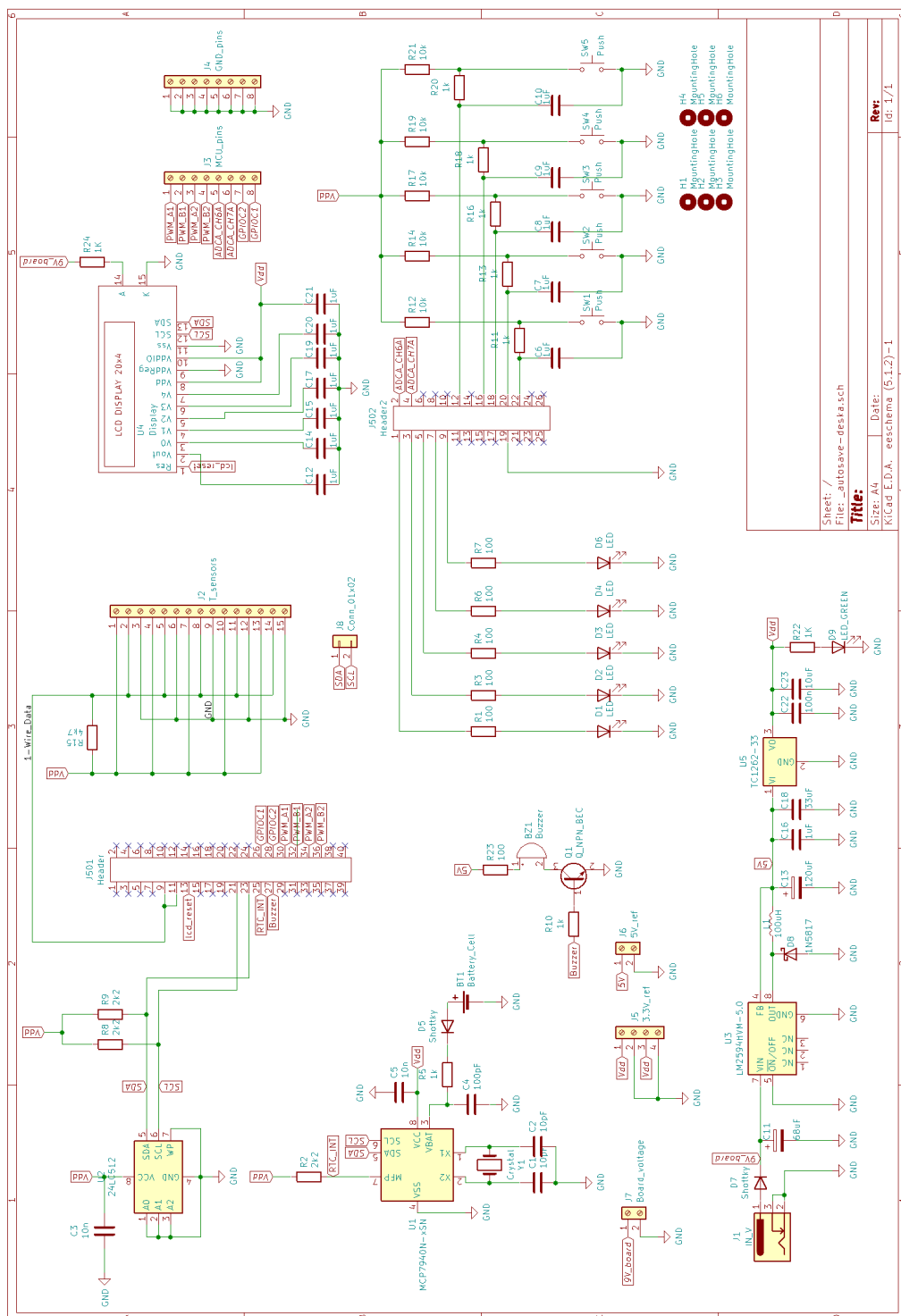
<b>Obr. 15: Time sloty pro čtení log. 0 a log. 1 [8]</b> .....	13
<b>Obr. 16: 64bitové číslo uložené v ROM [8]</b> .....	13
<b>Obr. 17: Vývojová deska TWR-KV46150M [9]</b> .....	14
<b>Obr. 18: Blokový diagram mikrokontroléru rodiny Kinetis V [10]</b> .....	14
<b>Obr. 19: Vnitřní schéma periferie eFlexPWM [10]</b> .....	16
<b>Obr. 20: Mechanismus časování PWM</b> .....	17
<b>Obr. 21: Blokové schéma I2C periferie [10]</b> .....	18
<b>Obr. 22: Schéma I2C sběrnice [16]</b> .....	19
<b>Obr. 23: Zapojení napájecí části</b> .....	21
<b>Obr. 24: Zapojení ovládací části</b> .....	22
<b>Obr. 25: Zapojení EEPROM paměti</b> .....	22
<b>Obr. 26: Zapojení RTC modulu</b> .....	23
<b>Obr. 27: Zapojení LCD displeje</b> .....	23
<b>Obr. 28: Zapojení 1-Wire sběrnice</b> .....	24
<b>Obr. 29: Zapojení pinů mikrokontroléru</b> .....	24
<b>Obr. 30: Osazena DPS, strana TOP</b> .....	25
<b>Obr. 31: Osazená DPS, strana BOTTOM</b> .....	25
<b>Obr. 32: Blokové schéma zařízení</b> .....	26
<b>Obr. 33: Status registr periferie PWM [10]</b> .....	27
<b>Obr. 34: Průběh reset pulzu 1-Wire sběrnice [12]</b> .....	29
<b>Obr. 35: Princip řízení 1-Wire sběrnice</b> .....	30
<b>Obr. 36: Stavový automat ovládání sběrnice</b> .....	30
<b>Obr. 37: Zápis logické 1 na sběrnici</b> .....	31
<b>Obr. 38: Čtení ze sběrnice</b> .....	32
<b>Obr. 39: Průběh na sběrnici – reset (vlevo) a presence (vpravo) pulz</b> .....	32
<b>Obr. 40: Stavový automat pro I2C sběrnici</b> .....	33
<b>Obr. 41: Průběh teploty uložené v paměti EEPROM v programu FreeMaster</b> .....	35
<b>Obr. 42: Ukázka funkčnosti zařízení</b> .....	36

## SEZNAM TABULEK

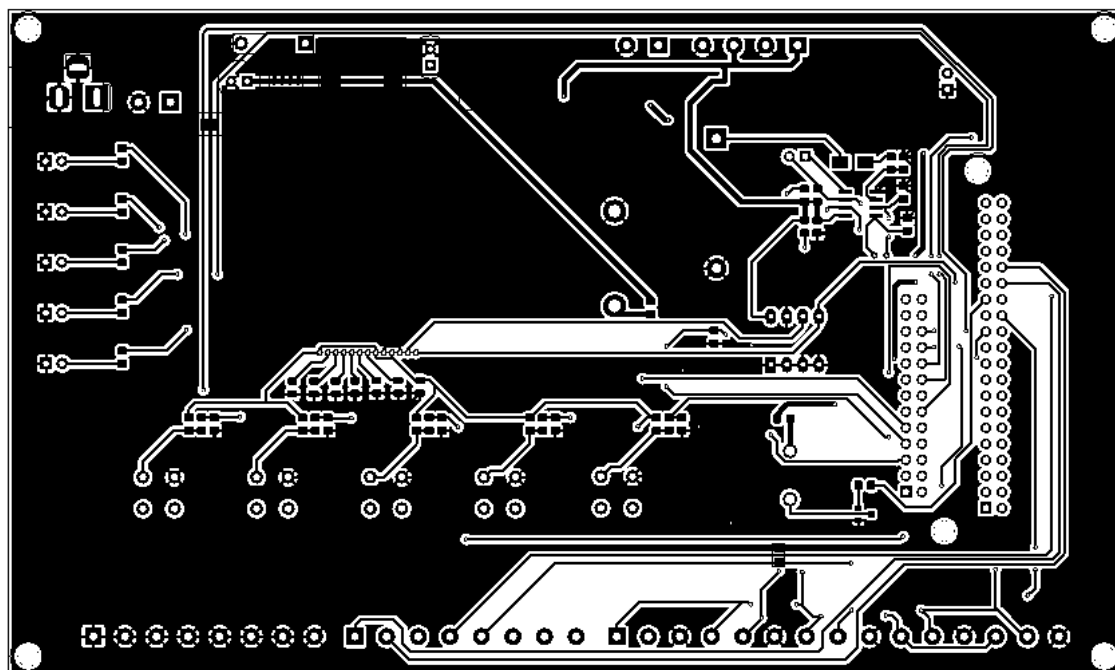
Tabulka 2: Srovnání teplotních senzorů .....	7
Tabulka 3: Výběr základních parametrů senzoru DS18B20 [8].....	8
Tabulka 4: ROM příkazy [8] .....	10
Tabulka 5: Funkční příkazy [8] .....	10

# A NÁVRH ZAŘÍZENÍ

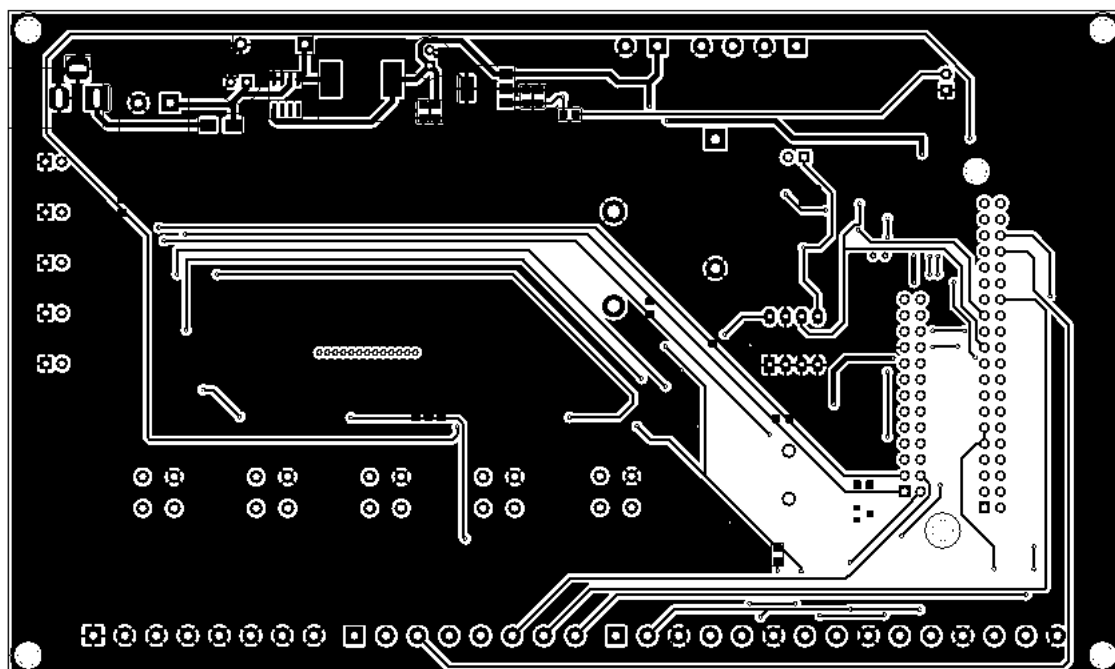
## A.1 Obvodové zapojení měřicí desky



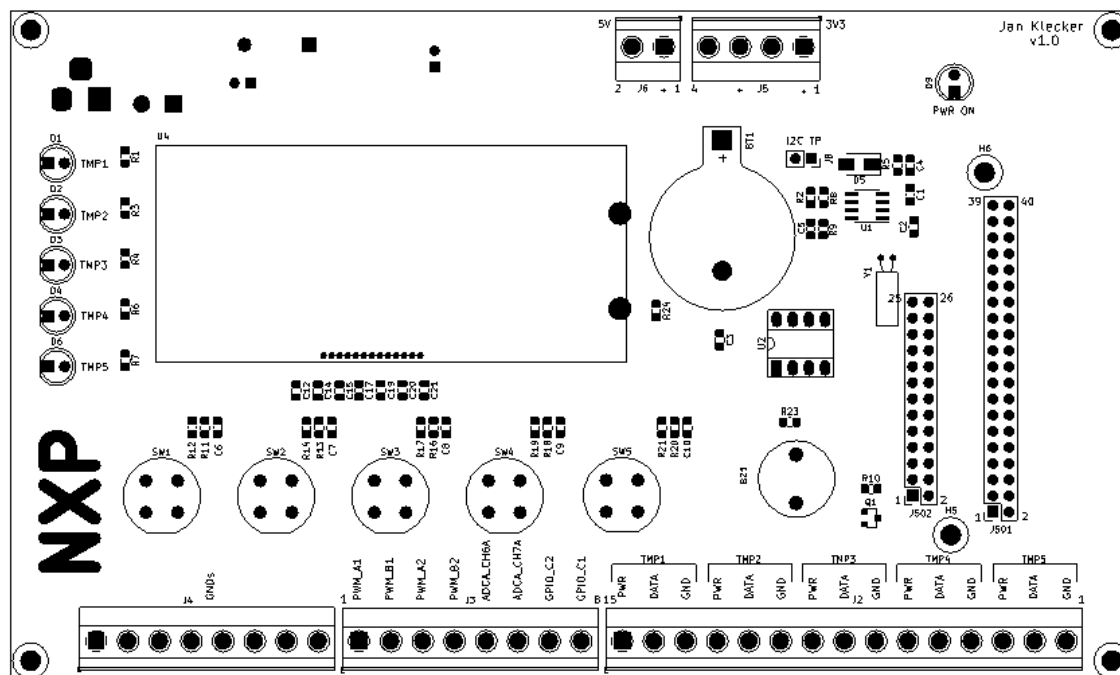
## A.2 DPS měřicí desky – strana TOP



## A.3 DPS měřicí desky – strana BOTTOM



## A.4 Osazovací plan – strana TOP



## A.5 Osazovací plán – strana BOTTOM

